A

Seminar report

On

8085 microprocessor

Submitted in partial fulfillment of the requirement for the award of degree
of Electronics

SUBMITTED TO: **SUBMITTED BY:**

www.studymafia.org   www.studymafia.org

Acknowledgement

I would like to thank respected Mr…….. and Mr. ……..for giving me such a wonderful opportunity to expand my knowledge for my own branch and giving me guidelines to present a seminar report. It helped me a lot to realize of what we study for.

Secondly, I would like to thank my parents who patiently helped me as i went through my work and helped to modify and eliminate some of the irrelevant or un-necessary stuffs.

Thirdly, I would like to thank my friends who helped me to make my work more organized and well-stacked till the end.

Next, I would thank Microsoft for developing such a wonderful tool like MS Word. It helped my work a lot to remain error-free.

Last but clearly not the least, I would thank The Almighty for giving me strength to complete my report on time.

Preface

I have made this report file on the topic **8085 microprocessor**; I have tried my best to elucidate all the relevant detail to the topic to be included in the report. While in the beginning I have tried to give a general view about this topic.

My efforts and wholehearted co-corporation of each and everyone has ended on a successful note. I express my sincere gratitude to …………..who assisting me throughout the preparation of this topic. I thank him for providing me the reinforcement, confidence and most importantly the track for the topic whenever I needed it.

INTRODUCTION TO 8085 MICROPROCESSOR

1.1 Introduction

Microprocessor is a Central Processing Unit (CPU) etched on a single chip. A single Integrated Circuit (IC) has all the functional components of a CPU namely Arithmetic Logic Unit (ALU), Control Unit and registers. The 8085 microprocessor is an 8-bit processor that includes on its chip most of the logic circuitry for performing computing tasks and for communicating with peripherals.

The architecture of a microprocessor is to be learnt in terms of registers, memory addressing, addressing modes, instruction set, interfacing with memory and Input and Output (I/O) devices and interrupt handling.

It is necessary to learn about the above mentioned concepts to write efficient assembly language programs, and to design microprocessor based systems. This unit gives you an overall idea about the microprocessors, the detailed discussion about 8085 architecture and interfacing of 8085 with Programmable Peripheral Interface (PPI) devices.

Learning Objectives

- To understand the basics and evolution of microprocessors
- To study about the functional components of 8085 in detail
- To discuss the different types of memory addressing schemes 0f 8085
- To learn the various addressing modes supported by 8085
- To study the various types of instructions provided by 8085
- To study the pin diagram and the signals of various pins of 8085
- To discuss about the timing and execution of instructions by 8085
- To understand the interrupt handling of 8085

Functional Components of a Microprocessor

A digital computer is a programmable machine specially designed for making computation. Its main components are: CPU (Central Processing Unit), memory, input device and output device as shown in figure 1.1.
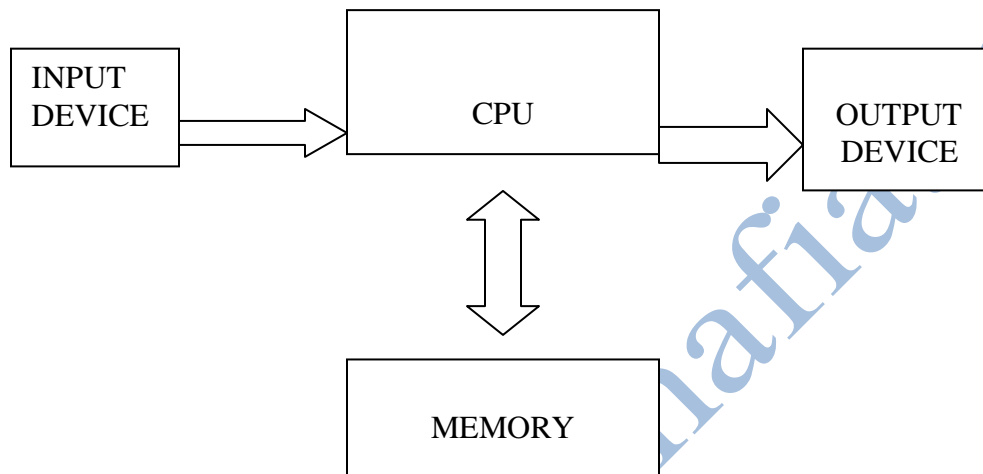

Figure 1.1 Schematic Diagram of a Digital Computer

A microcomputer is a small digital computer. The CPU of a microcomputer is a microprocessor. Other components are same as those of any other digital computer. In figure 1.1, if we change the label CPU as Microprocessor, we get the organization of a microcomputer.

The physical devices and circuitry of a computer are called hardware. A physical device may be electronic, magnetic, mechanical or an optical device etc. A sequence of instructions to perform a particular task is called a program. A set of programs written for a particular computer is known as software for that computer. The input and output devices are known as peripherals. Sometimes the term peripheral also includes memory. Programs are subroutines stored in ROM (Read Only Memory)s, Programmable ROM (PROM)s, Erasable PROM(EPROM)s and/or EEPROMs are known as firmware. The commonly available firmwares are: monitors, microprograms, subroutines for input and output devices.

The Central Processing Unit (CPU) fetches instructions from the memory and performs specified tasks. It stores results in the memory or sends results to the output device according to the instructions given in the program. The CPU controls

and communicates with memory and input/output devices. Under the control of the CPU, programs and data are stored in the memory and displayed on Cathode Ray Tube (CRT). The schematic diagram of a CPU is shown in figure 1.2.

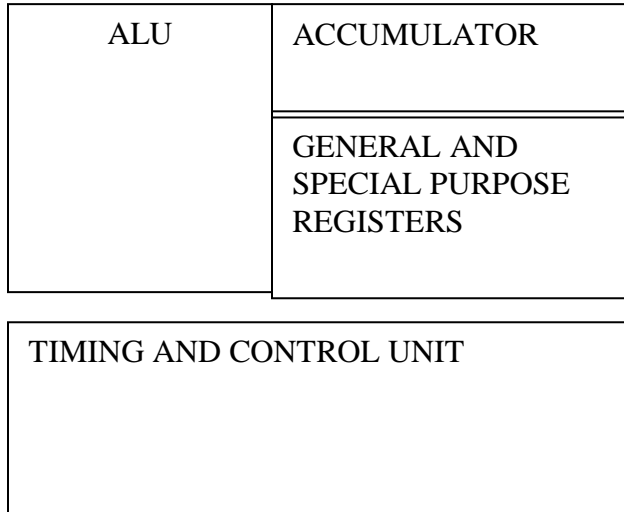| ALU | ACCUMULATOR |
| | GENERAL AND SPECIAL PURPOSE REGISTERS |
| TIMING AND CONTROL UNIT | |

Figure 1.2 Schematic Diagram of a CPU or a Microprocessor

The CPU of a large computer is implemented on one or more circuit boards. ICs are used as its components. Recent practice is to use microprocessors to perform different functions within the CPU of a large computer. The major sections of a CPU are Arithmetic and Logic Unit (ALU), Accumulator, General and Special purpose registers and Timing and Control Unit. The function of an ALU is to perform arithmetic operations such as addition and subtraction; and logical operations such as AND, OR and EXCLUSIVE-OR. Timing and control unit controls the entire operations of a computer. It acts as a brain. It also controls all other devices connected to the CPU. It generates timing signals necessary for input and output devices. The accumulator is a register, which contains one of the operands and stores results of most arithmetic and logical operations. General purpose registers are used for temporary storage of data and intermediate results while computer is making execution of a program. Special purpose registers are used by the microprocessor itself. Some of them are not accessible to programmers. Examples of special purpose registers are program counter, stack pointer, instruction register and status register.

The memory is a storage device. It stores program, data, results etc.

The computer receives data and instructions through input devices. An input device converts instructions, input data and signals into proper binary form suitable for a digital computer. A key-board and simple switches are used as input devices. The user enters instructions and data through a key-board or simple switches. Computers are also used to measure and control physical quantities like temperature, pressure, speed, position etc. For these purposes transducers are used to convert physical quantities into proportional electrical signals. A/D converters are used to convert analog electrical signals into digital signals, which are sent to the computer. Transducers and sensors, data acquisition system etc. are also included in input devices. A/D converter forms a part of data acquisition system.

The computer sends results to output devices. An output device may store, print, display or send electrical signal to control/actuate certain equipment. The examples of simple output devices are printers, CRT, LEDs, D/A converter, controllers, actuators etc. Sometimes input and output devices may be combined in a single unit, which acts as both an input as well as an output device. A keyboard and CRT are combined to form a video terminal, which is a common I/O device for human interaction with a computer.

With the advent of LSI and VLSI technology it became possible to build the entire CPU on a single chip IC. A CPU built into a single LSI/VLSI chip is called a microprocessor. A digital computer using microprocessor as its CPU is called a microcomputer. The term micro initiates its physical size; not it's computing power. Today the computing power of a powerful microprocessor approaches that a CPU on earlier large computer. The main sections of a microprocessor are: ALU, timing and control unit, accumulator, general purpose and special purpose registers. In this subject we'll study about two microprocessors namely Intel 8085 (8-bit) and Intel 8086 (16-bit).

Have you understood?

1. What are the major components of a digital computer?
2. What are the functional components of a CPU?
3. What is a microprocessor?

Evolution of Microprocessors

The first microprocessor was introduced in 1971 by Intel Corporation, U.S.A. It was a 4-bit microprocessor, the Intel 4004. The 4004 was introduced on November 15, 1971 and originally ran at a clock speed of 108KHz (108,000 cycles per second, or just over one-tenth a megahertz). The 4004 contained 2,300 transistors and was built on a 10-micron process. This means that each line, trace, or transistor could be spaced about 10 microns (millionths of a meter) apart. Data was transferred 4 bits at a time, and the maximum addressable memory was only 640 bytes. The 4004 was designed for use in a calculator but proved to be useful for many other functions because of its inherent programmability. In 1972, Intel introduced the 1st 8-bit processor, the Intel 8008. The Intel 8004 and 8008 both used Positive Channel Metal Oxide Semiconductor (PMOS) technology. In 1973 a more powerful and faster 8-bit processor, the Intel 8080 was introduced. It employed Negative Channel metal Oxide semiconductor (NMOS) technology. The 8008 processor contained 3,500 transistors and was built on the same 10-micron process as the previous processor. The big change in the 8008 was that it had an 8-bit data bus, which meant it could move data 8 bits at a time twice as much as the previous chip. It could also address more memory, up to 16KB. This chip was primarily used in dumb terminals and general-purpose calculators.

The next chip in the lineup was the 8080, introduced in April 1974, running at a clock rate of 2MHz. Due to mostly the faster clock rate, the 8080 processor had 10 times the performance of the 8008. The 8080 chip contained 6,000 transistors and was built on a 6-micron process. Similar to the previous chip, the 8080 had an 8-bit data bus, so it could transfer 8 bits of data at a time. The 8080 could address up to 64KB of memory, significantly more than the previous chip. It was the 8080 that helped start the PC revolution because this was the processor chip used in what is generally regarded as the first personal computer, the Altair 8800. The CP/M operating system was written for the 8080 chip, and Microsoft was founded and delivered its first product: Microsoft BASIC for the Altair. These initial tools

provided the foundation for a revolution in software because thousands of programs were written to run on this platform. In fact, the 8080 became so popular that it was cloned. A company called Zilog formed in late 1975, joined by several ex-Intel 8080 engineers. In July 1976, it released the Z-80 processor, which was a vastly improved version of the 8080. It was not pin compatible but instead combined functions such as the memory interface and RAM refresh circuitry, which enabled cheaper and simpler systems to be designed. The Z-80 also incorporated a superset of 8080 instructions, meaning it could run all 8080 programs. It also included new instructions and new internal registers, so software designed for the Z-80 would not necessarily run on the older 8080. The Z-80 ran initially at 2.5MHz (later versions ran up to 10MHz) and contained 8,500 transistors. The Z-80 could access 64KB of memory.

Intel released the 8085, it's follow-up to the 8080, in March 1976. Even though it predated the Z-80 by several months, it never achieved the popularity of the Z-80 in personal computer systems. It was popular as an embedded controller, finding use in scales and other computerized equipment. The 8085 ran at 5MHz and contained 6,500 transistors. It was built on a 3-micron process and incorporated an 8-bit data bus. Along different architectural lines, MOS Technologies introduced the 6502 in 1976. This chip was designed by several ex-Motorola engineers who had worked on Motorola's first processor, the 6800. The 6502 was an 8-bit processor like the 8080, but it sold for around $25, whereas the 8080 cost about $300 when it was introduced. The price appealed to Steve Wozniak, who placed the chip in his Apple I and Apple II designs. The chip was also used in systems by Commodore and other system manufacturers. The 6502 and its successors were also used in game consoles, including the original Nintendo Entertainment System (NES) among others. Motorola went on to create the 68000 series, which became the basis for the Apple Macintosh line of computers. Today those systems use the PowerPC chip, also by Motorola and a successor to the 68000 series.

All these previous chips set the stage for the first PC processors. Intel introduced the 8086 in June 1978. The 8086 chip brought with it the original x 86 instructions set that is still present in current x86-compatible chips such as the Pentium 4 and AMD Athlon. A dramatic improvement over the previous chips, the 8086 was a full 16-bit design with 16-bit internal registers and a 16-bit data bus. This meant that it could work on 16-bit numbers and data internally and also transfer 16 bits at a time in and out of the chip. The 8086 contained 29,000 transistors and initially ran at up to 5MHz. The chip also used 20-bit addressing, so it could directly address up to 1MB of memory. Although not directly backward compatible with the 8080, the 8086 instructions and language were very similar and enabled older

programs to quickly be ported over to run. This later proved important to help jumpstart the PC software revolution with recycled CP/M (8080) software.

Although the 8086 was a great chip, it was expensive at the time and more importantly required expensive 16-bit board designs and infrastructure to support it. To help bring costs down, in 1979 Intel released what some called a crippled version of the 8086 called the 8088. The 8088 processor used the same internal core as the 8086, had the same 16-bit registers, and could address the same 1MB of memory, but the external data bus was reduced to 8 bits. This enabled support chips from the older 8-bit 8085 to be used, and far less expensive boards and systems could be made. These reasons are why IBM chose the 8088 instead of the 8086 for the first PC. This decision would affect history in several ways. The 8088 was fully software compatible with the 8086, so it could run 16-bit software. Also, because the instruction set was very similar to the previous 8085 and 8080, programs written for those older chips could be quickly and easily modified to run. This enabled a large library of programs to be quickly released for the IBM PC, thus helping it become a success. The overwhelming blockbuster success of the IBM PC left in its wake the legacy of requiring backward compatibility with it. To maintain the momentum, Intel has pretty much been forced to maintain backward compatibility with the 8088/8086 in most of the processors it has released since then.

To date, backward compatibility has been maintained, but innovating and adding new features has still been possible. One major change in processors was the move from the 16-bit internal architecture of the 286 and earlier processors to the 32-bit internal architecture of the 386 and later chips, which Intel calls IA-32 (Intel Architecture, 32-bit). Intel's 32-bit architecture dates to 1985, and it took a full 10 years for both a partial 32-bit mainstream OS (Windows 95) as well as a full 32-bit OS requiring 32-bit drivers (Windows NT) to surface, and another 6 years for the mainstream to shift to a fully 32-bit environment for the OS and drivers (Windows XP). That's a total of 16 years from the release of 32-bit computing hardware to the full adoption of 32-bit computing in the mainstream with supporting software.

Now we are in the midst of another major architectural jump, as Intel and AMD are in the process of moving from 32-bit to 64-bit computing for servers, desktop PCs, and even portable PCs. Intel had introduced the IA-64 (Intel Architecture, 64-bit) in the form of the Itanium and Itanium 2 processors several years earlier, but this standard was something completely new and not an extension of the existing 32-bit technology. IA-64 was first announced in 1994 as a CPU development project with Intel and HP (codenamed Merced), and the first technical details were

made available in October 1997. The result was the IA-64 architecture and Itanium chip, which was officially released in 2001. The fact that the IA-64 architecture is not an extension of IA-32 but is instead a whole new and completely different architecture is fine for non-PC environments such as servers, but the PC market has always hinged on backward compatibility. Even though emulating IA-32 within IA-64 is possible, such emulation and support is slow.

With the door now open, AMD seized this opportunity to develop 64-bit extensions to IA-32, which it calls AMD64 (originally known as x86-64). Intel eventually released its own set of 64-bit extensions, which it calls EM64T or IA-32e mode. As it turns out, the Intel extensions are almost identical to the AMD extensions, meaning they are software compatible. It seems for the first time that Intel has unarguably followed AMD's lead in the development of PC architecture. To make 64-bit computing a reality, 64-bit operating systems and 64-bit drivers are also needed. Microsoft began providing trial versions of Windows XP Professional x64 Edition (which supports AMD64 and EM64T) in April 2005, and major computer vendors now offer systems with Windows XP Professional x64 already installed. Major hardware vendors have also developed 64-bit drivers for current and recent hardware. Linux is also available in 64-bitcompatible versions, making the move to 64-bit computing possible.

The latest development is the introduction of dual-core processors from Intel, IBM, Sun and AMD. Dual-core processors have two full CPU cores operating off of one CPU package in essence enabling a single processor to perform the work of two processors. Although dual-core processors don't make games (which use single execution threads and are usually not run with other applications) play faster, dual-core processors, like multiple single-core processors, split up the workload caused by running multiple applications at the same time. If you've ever tried to scan for viruses while checking email or running another application, you've probably seen how running multiple applications can bring even the fastest processor to its knees. With dual-core processors available from both Intel and AMD, your ability to get more work done in less time by multitasking is greatly enhanced. Current dual-core processors also support AMD64 or EM64T 64-bit extensions, enabling you to enjoy both dual-core and 64-bit computing's advantages.

PCs have certainly come a long way. The original 8088 processor used in the first PC contained 29,000 transistors and ran at 4.77MHz. The AMD Athlon 64FX has more than 105 million transistors, while the Pentium 4 670 (Prescott core) runs at 3.8GHz and has 169 million transistors thanks to its 2MB L2 cache. Dual-core processors, which include two processor cores and cache memory in a single

physical chip, have even higher transistor counts: The Intel Pentium D processor has 230 million transistors, and the AMD Athlon 64 X2 includes over 233 million transistors. As dual-core processors and large L2 caches continue to be used in more and more designs, look for transistor counts and real-world performance to continue to increase. And the progress doesn't stop there because, according to Moore's Law, processing speed and transistor counts are doubling every 1.52 years.

INTEL 8085

Intel 8085 is an 8-bit, N-channel Metal Oxide semiconductor (NMOS) microprocessor. It is a 40 pin IC package fabricated on a single Large Scale Integration (LSI) chip. The Intel 8085 uses a single +5V DC supply for its operation. Its clock speed is about 3MHz. The clock cycle is of 320 ns. The time for the clock cycle of the Intel 8085 is 200 ns. It has 80 basic instructions and 246 opcodes. The 8085 is an enhanced version of its predecessor, the 8080A; its instruction set is upward compatible with that of the 8080A, meaning that 8085 instruction set includes all the 8080A instructions plus some additional ones. Programs written for 8080A will be executed by 8085, but the 8085 and 8080A are not pin compatible.

Architecture

The architecture of Intel 8085 consists of three main sections, arithmetic and logic unit, timing and control unit and several registers. The functional block diagram of 8085 is shown in figure 1.3. These important sections are described in the subsequent sections.

Arithmetic and Logic Unit (ALU)

The ALU performs the following arithmetic and logical operations.
1. Addition
2. Subtraction
3. Logical AND
4. Logical OR
5. Logical EXCLUSIVE OR
6. Complement (logical NOT)
7. Increment (add 1)
8. Decrement (subtract 1)
9. Left shift
10. Clear

The ALU is the unit that manipulates the data. ALU includes the accumulator, the temporary register, the arithmetic and logic circuits and flags.
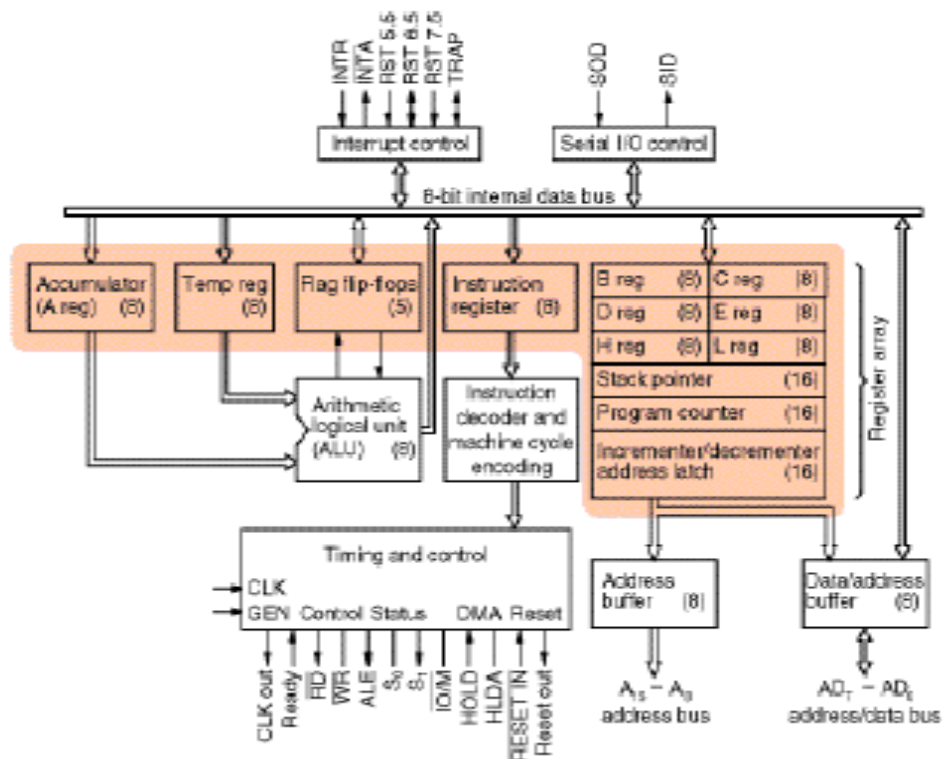
Figure 1.3 Architecture of 8085

Timing and Control Unit

The timing and control unit is a section of the CPU. It generates timing and control signals, which are necessary for the execution of instructions. It controls data flow between CPU and peripherals (including memory). It provides status, control and timing signals, which are required for the operation of memory and I/O devices. It controls the entire operations of the microprocessor and peripherals connected to it. Hence you can understand that the control unit of the CPU acts as the brain of a computer system. This unit synchronizes all the microprocessor operations with the clock and generates the control signals necessary for communication between the microprocessor and peripherals.

Registers

Figure 1.3 shows the various registers of Intel 8085.

Registers are used by the microprocessor for temporary storage and manipulation of data and instructions. Data remain in the register till they are sent to the memory or I/O devices. In a large computer the number of registers is more and hence the program requires less transfer of data to and from the memory. In a small computer the number of registers is small due to the limited size of the chip. If a digital computer requires frequent access to memory then the performance comes down due to the mismatch in the speed with which the CPU and the memory operate. The 8085 programming model includes six registers, one accumulator, and one flag register, as shown in Figure. In addition, it has two 16-bit registers: the stack pointer and the program counter. They are described briefly as follows.

General Purpose Registers

The 8085 has six general-purpose registers to store 8-bit data; these are identified as B, C, D, E, H, and L as shown in the figure 1.4. They can be combined as register pairs - BC, DE, and HL - to perform some 16-bit operations. The programmer can use these registers to store or copy data into the registers by using data copy instructions. The HL register pair is also used to address memory locations. In other words, HL register pair plays the role of memory address register.

Accumulator

The accumulator is an 8-bit register that is a part of arithmetic/logic unit (ALU). This register is used to store 8-bit data and to perform arithmetic and logical operations. The result of an operation is stored in the accumulator. The accumulator is also identified as register A. This is used during the execution of a program for temporary storage. It holds one of the operands, which serves as one of the inputs to ALU. The other operand may be either in the memory or in one of the registers. The final result of an arithmetic or logic operation is placed in the accumulator.
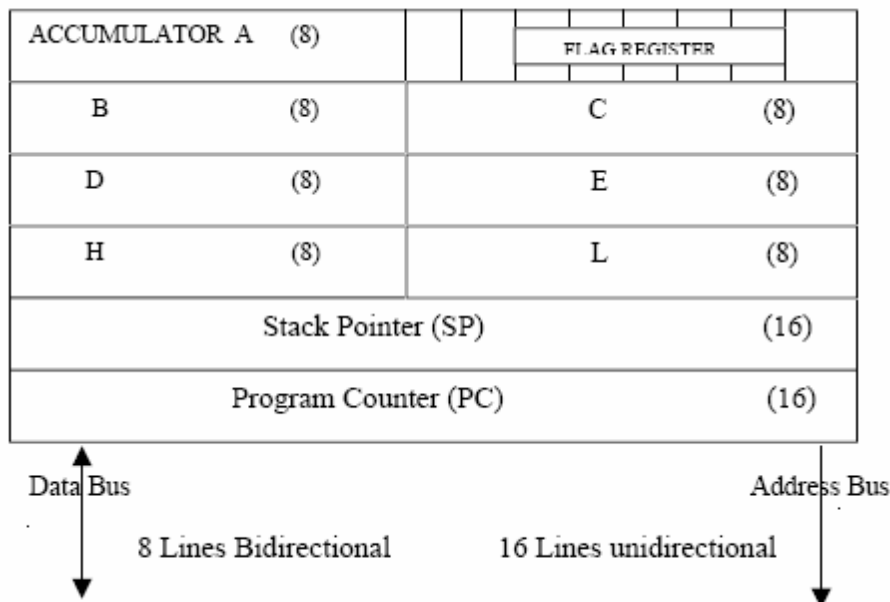
Figure 1.4 General Registers

Other Registers

Program Counter

This 16-bit register deals with sequencing the execution of instructions. This register is a memory pointer. Memory locations have 16-bit addresses, and that is why this is a 16-bit register. The microprocessor uses this register to sequence the execution of the instructions. The function of the program counter is to point to the memory address from which the next byte is to be fetched. When a byte (machine code) is being fetched, the program counter is incremented by one to point to the next memory location. However, please note that the program counter is loaded with some absolute value during the execution of branch instructions.

Stack Pointer

The stack pointer is also a 16-bit register used as a memory pointer. It points to a memory location in R/W memory, called the stack. The beginning of the stack is defined by loading 16-bit address in the stack pointer. The stack is the sequence of memory locations defined by the programmer. The stack is used to save the content of a register during the execution of the program.

Instruction Register/Decoder

The instruction register and the decoder are considered as a part of the ALU. It is a temporary storage for the current instruction of a program. Latest instruction is sent here from memory prior to execution. The decoder decodes the instruction and establishes the sequence of events to follow. However, you please understand that the instruction register is not programmable and can not be accessed through any instruction.

Flag Register

The ALU includes five flip-flops, which are set or reset after an operation according to data conditions of the result in the accumulator and other registers. They are called Zero (Z), Carry (CY), Sign (S), Parity (P), and Auxiliary Carry (AC) flags; their bit positions in the flag register are shown in the Figure below. The most commonly used flags are Zero, Carry, and Sign. The microprocessor uses these flags to test data conditions. The flag register is shown in figure 1.5.
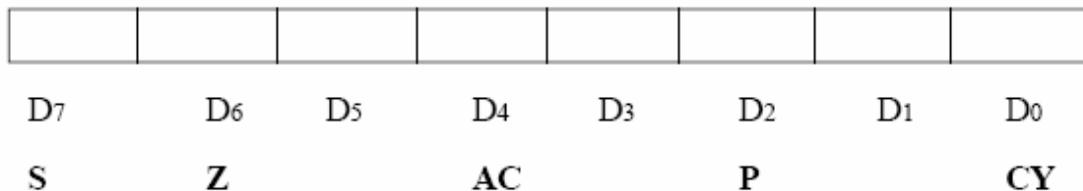
D7		D6	D5		D4	D3		D2	D1		D0
S		Z			AC			P			CY

Figure 1.5 Flag Register of 8085

For example, after an addition of two numbers, if the sum in the accumulator id larger than eight bits, the flip-flop uses to indicate a carry -- called the Carry flag (CY) – is set to one. When an arithmetic operation results in zero, the flip-flop called the Zero (Z) flag is set to one. The first Figure shows an 8-bit register, called the flag register, adjacent to the accumulator. However, it is not used as a register; five bit positions out of eight are used to store the outputs of the five flip-flops. The flags are stored in the 8-bit register so that the programmer can examine these flags (data conditions) by accessing the register through an instruction.

These flags have critical importance in the decision-making process of the microprocessor. The conditions (set or reset) of the flags are tested through the software instructions. For example, the instruction JC (Jump on Carry) is implemented to change the sequence of a program when CY flag is set. The

thorough understanding of flag is essential in writing assembly language programs. The combination of the flag register and the accumulator is called Program Status Word (PSW) and PSW is the 16-bit unit for stack operation.

Features

- 8 bit microprocessor(8085 microprocessor can read or write or perform arithmetic and logical operations on 8-bit data at time)
- It has 8 data lines and 16 address lines hence capacity is 216 = 64 kB of memory
- Cock frequency is 3 MHz
- It requires +5V power supply.
- It is a single chip NMOS device implemented with 6200 transistors.
- It provides 74 instructions with five addressing modes.
- It provides 5 hardware interrupt and 8 software interrupts.

REFERENCES

1. www.google.com

2. www.wikipedia.org

3. www.studymafia.org

4. www.pptplanet.com