ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

Comparative Studies on Intel 8085 Microprocessor and Intel 8051 Microcontroller With Some Suitable Assembly Language Programs

U. Sridevi , Dr. D. Chinni Krishna

Lecturer & Head, Dept. of Mathematics, Govt. City College, Nayapul, Hyderabad, Telangana State, India

Senior Lecturer in Physics, Dept. of Physics, Bhavan's New Science College (PG), Narayanaguda, Hyderabad, Telangana State, India,

ABSTRACT: Microprocessors are becoming an essential part of human life. Knowingly are unknowingly everyone is using the services of microprocessors because in almost all electronic home appliances like Air conditioners, Refrigerators, Washing machines, Mobile phones, Internet, microwave ovens etc., microprocessors are involved. The most useful and familiar microprocessor is 8085 microprocessor. The most useful processor for control applications is 8051 microcontroller. Although both are 8 bit processor, they possess many similarities and lot of differences in architectural, instructional and addressing modes point of view and also in writing and executing the assembly language programs. Therefore a systematic analysis was carried out to explain the similarities and the differences of 8085 & 8051in all the above aspects.

Compare with the 8085 microprocessor, 8051 microcontroller has more number of registers and more special functional registers (SFRs). The superior quality of microcontroller is the accessibility of individual bits in many of the special function registers and all four ports of 8051 microcontroller. The 128 bytes of on chip RAM and 128 bytes of external RAM is playing an important role in the 8051 microcontroller. For loading and transferring the data into the registers only MOV instruction is used in 8051 microcontroller, where as MVI, LXI instructions are used for loading the data and MOV instruction transfers the data in 8085 microprocessors. The MUL, DIV and SWAP instructions are the advanced instructions available in 8051. It is mandatory to push a pair of registers in 8085 microprocessors where as the individual registers can be pushed in to the stack in 8051 microcontroller by using direct addressing mode. Because of DAD instruction, the double addition is more comfort with 8085 microprocessor than 8051 microcontroller, but, the availability of MUL & DIV instructions, executing multiplication and division programs with 8051are easier than in 8085 microprocessor.

KEYWORDS: Microprocessors, Microcontrollers, Registers, Assembly Language Program, Double Addition, Architecture and Instruction.

1. INTRODUCTION

INTRODUCTION TO MICROPROCESSORS: Microprocessor is an electronic device used to process the data from memory to registers or register to register or from memory to outside world and vice versa within microseconds of time. It is a multipurpose programmable logic device that reads binary instructions from a storage device called memory, accepts binary data as input and possesses data according to those instructions and provides results as output. The microprocessor operates in binary digits 0 and 1 also known as bits. These digits are represented in terms of electrical voltages (GND and 5Volts) in the machine as shown in figure 1.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

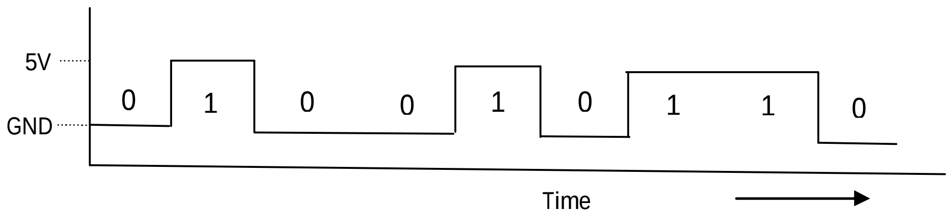

Fig.1. Digital Voltage levels in 0s and 1s format

Each microprocessor recognizes and processes a group of 8 bits (byte) called the word. A processor with an 8-bit word is known as an 8-bit processor. The microprocessor can be divided into 3 segments as shown in Fig.2.

ARITHMETIC AND LOGICAL UNIT (ALU): It is the area of the microprocessor where various computing functions are performed on data. The ALU performs arithmetic and logical operations and store the results either in registers or in memory.

REGISTER ARRAY: It consists of various registers. These registers primarily used to store data temporarily during the execution of a program. Some of the registers are accessible to the user through instructions.

CONTROL UNIT: It provides the necessary timing and control signals to all the operations in the microprocessors. It controls the flow of data between the microprocessor and memory and peripherals.

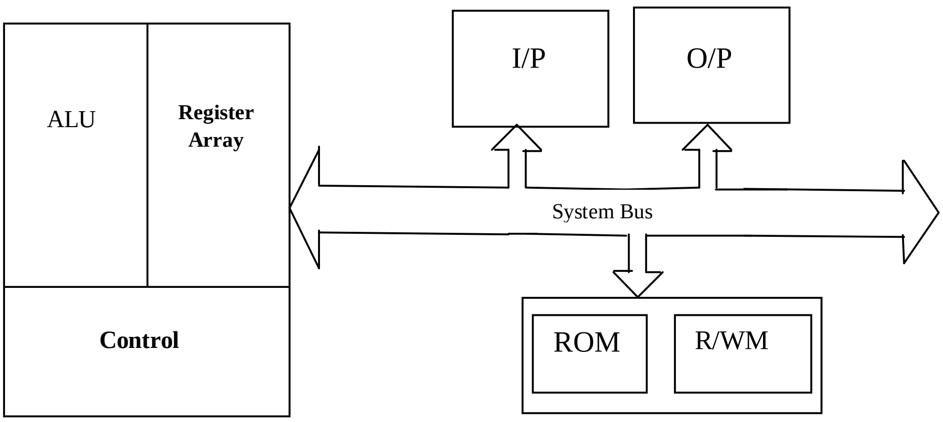

Fig.2. Block diagram of a general purpose microprocessor

INTRODUCTION TO 8085 MICROPROCESSORS: The Intel 4004 was the first 4 bit PMOS programmable device, which was introduced in the year 1971 and that was primarily used in calculators. It was quickly replaced by the 8 bit microprocessor (the Intel 8008), which was in turn superseded by Intel 8080. In mid 1970s, the Intel 8080 was widely used in control applications and small computers also were designed using the 8080 as the CPU. These computers become known as microcomputers. Within a few years after the emergence of 8080, the Motorola 6800, the Zilog Z80 and the Intel 8085 microprocessors were developed as improvements over the 8080(1). As the microprocessor began to acquire more and more computing functions, they were viewed more as CPUs rather than as programmable logic devices. The 8 bit microprocessors are not simply being replaced by more powerful microprocessors; most microcomputers are now built with 32 and 64 bit microprocessors. However, each microprocessor has begun to carve a role for its own applications. The 8 bit microprocessors are being used as programmable logic devices in control applications and more powerful microprocessors are being used for mathematical computing, data processing and computer graphics applications.

INTRODUCTION TO 8051 MICROCONTROLLERS: In 1976, Intel introduced the 8048 series of single chip microcontrollers. It is also known as MCS-48(2). The microcontrollers of this series are: 8048, 8041, 8042, 8049 and

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

8035. These microcontrollers were popular and widely used for automatic control applications. These microcontrollers do not have multiplication and division instructions. In 1981, Intel Corporation introduced a more powerful 8-bit microcontroller called the 8051, generally referred as MCS-51. These are the second generation microcontrollers. It has 128 bytes of RAM, 4K bytes of on chip ROM, two timers, one serial port and four ports (each 8 bits wide) all on a single chip. The 8051 is an 8-bit processor, meaning that the CPU can work on only 8 bits of data at a time. Data larger than 8 bits has to be broken in to 8 bit pieces to be processed by the CPU [(3)]. The total 256 data's are appeared in Hexadecimal digits in the range from 00–FFH [(4)]. The 8051 has became widely popular after Intel allowed other manufacturers to make and market any flavour of the 8051 they please with the condition that they remain code compatible with the 8051. Although the 8051 is the most popular member of the 8051 family, it is not appear in the part number, because it is available in different memory types such as UV-EPROM version of the 8051 is the 8751, the flash ROM version is marketed by many companies including Atmel Corp. The Atmel Flash is called AT89C51. The NV-RAM version of the 8051 made by Dallas Semiconductor is called DS5000. The AT89C51-12PC, where C before 51 is for CMOS, which has a low power consumption, 12 indicates 12 MHz, P is for plastic DIP package and C for commercial (Vs. M for Military) is ideal for many student projects and research work.

II. EXPERIMENTAL TECHNIQUES

The assembly language programs of 8085 were executed on MPS 85-3 trainer kit[(5)] which is based on 8085 CPU and 32K bytes of CMOS static RAM are provided with battery backup option. Total on board memory can be up to 64K Bytes. The 8085 is operated at 3.072 MHz and the microcontroller used in the present work is the SDA 51-MEL is a system design aid for operation of microcontroller devices[(6)]. It uses 8031/51 as the controller. It is a powerful microcontroller trainer for the INTEL 8051 microcontroller and it is operating at 11.0592 MHz.

III. RESULTS AND DISCUSSIONS

THE ARCHITECTURAL SIMILARITIES AND DIFFERENCES OF 8085 AND 8051: Microprocessor is becoming an essential part of human life like electric power. In all most all electrical appliances like Refrigerator, Air conditioner, Mobile Phones, Televisions, Washing machines, Microwave ovens etc are controlled by microprocessors. The most useful and popular microprocessor is 8085 and the popular microcontroller is 8051.

Table 1. Similarities and the differences of 8085 microprocessors and 8051 microcontroller

S. No.	In 8085 microprocessors	In 8051 microcontroller
1	RAM, ROM, timers, serial port and four ports (each 8 bits wide) all are connected externally to the CPU. Therefore it is a general purpose microprocessor	CPU, 128 bytes of RAM, 4K bytes of on chip ROM, two timers, one serial port and four ports (each 8 bits wide) all on a single chip. Therefore it a microcontroller.
2	8- bit processor	8- bit processor
3	The total addressing capability of a CPU is 2^{16} bytes = 64 Kilo bytes, 1 Kilo = 1024 in computers. The address locations are from 0000 to FFFFH	The total addressing capability of a CPU is 2^{16} bytes = 64 Kilo bytes, 1 Kilo = 1024 in computers. The address are from 0000 to FFFFH
4	8-bit Accumulator, which is associated with ALU. Bit accessible registers are not available in 8085.	8-bit Accumulator is closely associated with ALU. Its location is E0H of RAM. It is also a bit accessible register
5	Six 8-bit general purpose registers. They are B, C, D, E, H and L. To hold a 16-bit data a combination of two 8-bit registers can be employed. The valid register pairs are BC, DE and HL. The system monitor utilizes RAM locations 8F90H to 8FFFH for Stack and variables.	It has four banks of registers and each bank has eight 8-bit registers R0, R1, R2, R3, R4, R5, R6 and R7. Data bits D4 and D3 of PSW are used to select the desired register bank. First 32 bytes of 128 bytes of RAM are set aside for the register banks and the stack.
6	Size of Program Counter is 16 bits. It keeps track of memory addresses of the instructions in a program.	Size of Program Counter is 16 bits. It keeps track of memory addresses of the instructions in a program.
7	Size of Stack Pointer is 16 bits. Generally highest addresses of the memory can be used as stack.	Size of Stack Pointer is 8 bits. By default, 07^{th} location of RAM is acts as SP. 30H-7FH of RAM can be used as stack.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

8	The HL pair is used to act as memory pointer and it holds the 16 bit address of a memory location	The 16 bit data pointer (DPTR) to hold 16-bit address. It consists of DPH and DPL to hold 8-bit data.
9	The five status flags are available in 8 bit Flag register as shown below. They are Carry, Parity, Axiallary Carry, Zero and Sign Flags. S Z X AC X P X CY Program status word is the combination of Flag register and Accumulator. It is a 16 bit register and it is used to store the flag conditions in the Stack memory.	8-bit Program status word (PSW) register contains program status information as shown below: CY AC F0 RS1 RS0 OV -- P Only 4 flag bits Carry, Axiallary Carry, Overflow and Parity flags and RS1&RS0 bits are used to select the bank of registers. PSW is a bit accessible register and its bits are accessed as PSW.0 – PSW.7
10	Instruction register holds the opcode of the instruction which is decoded and executed. Temporary register associated with ALU holds data during an arithmetic / logical operation. It is used by the processor and not accessible to programmer.	There are number of special function registers (SFRs). The locations of SFRs are from 80H – FFH of RAM. A, B, PSW and DPTR are part of SFRs. The SFRs can be accessed by their names or by their addresses. Some of the SFRs have the capability of single bit accessibility.
11	**Interrupts:** The five interrupts with their service routines on their priority are TRAP → 0024H; RST 7.5→ 003CH; RST 6.5→ 0034H; RST 5.5→ 002C, INTR→ It is sampled by the microprocessor in the last state of the last machine cycle of each instruction. The RST n instructions are used for software interrupt.	The six interrupts and their interrupt service routines on their priority are as follows: Reset → 0000H, External hardware interrupt 0 (INT0) → 0003H Timer 0 interrupt (TF0) → 000BH, External hardware interrupt 1 (INT1) → 0013H, Timer 1 interrupt (TF1) → 001BH, Serial COM interrupt (RI and TI) → 0023H
12	The device has 40 pins requires +5V single power supply, operate with a 3 MHz single phase clock. All signals classified in 6 groups(i) Address bus (ii) Multiplexed data bus (iii) Control and status signals (iv) Power supply&frequency signals (v) External initiated signals and (vi) Serial input and output ports.	Available in 40 pin, in which 32 pins are reserved for 4 multiplexed address and data ports (Port 0. Port 1, Port 2 and Control port) and 2 pins for power supply, 2 pins for XTAL oscillator, 1 ALE pin to select address or data through ports, $\overline{EA}$ for external access, $\overline{PSEN}$ for external program memory and RST for resetting the device

There are several books available in the market to give deep subject about 8085 microprocessors and also for 8051 microcontrollers, but there are no sufficient material is available to compare the functions and their performances of 8085 microprocessors and 8051 microcontroller at a single flat form. Therefore, a systematic analysis was carried out by writing and executing some assembly language programs of both the processors to verify the functions and their performances of both the processors. The architectural similarities and differences of 8085 microprocessors and 8051 microcontroller were shown in Table 1.

A powerful feature of 8051 I/O ports is their capability to access individual bits of the port without altering the rest of the bits in that port. There are 16 bytes in the RAM from 20H to 2FH are reserved for single bit addressability. The individual bits of these bytes can be accessed as 00H – 7FH. On far with regular registers, there are several special function registers available in 8051. They are DPTR, IP, IE, TMOD, TCON, Two 16 bit timers, SCON, SBUF and PCON etc.; In 8051 the data pointer register can be used as a memory pointer to access data from external ROM. The instruction used is MOVX A,@DPTR, where X stands for external ROM. The DPTR register can accessed as DPH and DPL. There are two 16 bit timers Timer 0 and Timer 1 are used to generate time delays or to count the events happening in the processor (event counter). The timer 1 register can be accessed as TH1 and TL1, timer 0 register can be accessed as TH0 and TL0. An interrupt is an external or an internal event that interrupts the CPU to inform that a device needs its service. The program which is associated with the interrupt is called interrupt service routine. Various interrupts used by the 8085 microprocessor and 8051 microcontroller were also given in Table 1. Because of space limitations, complete analysis of instructions, timing analysis and the similarities and differences in interfacing concepts with various devices with 8085 microprocessor and 8051 microcontroller is not undertaken in the present work.

THE INSTRUCTIONAL SIMILARITIES AND DIFFERENCES OF 8085 AND 8051: To compare the similarities and the differences of assembly language instructions of both 8085 and 8051 a list of major instructions were shown in Table 2. It is observed from the Table 2 that, the instruction MVI instruction is used to load all the 8 bit registers of

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

8085 microprocessor, but LXI instruction is used to load the pair of registers. The MOV instruction is used to transfer the data's from registers to registers and memory to registers and vice versa. The MOV instruction is used to load as well as to transfer data in between the registers and from memory to registers and vice versa in 8051 microcontroller. The instruction LDA 8050H of 8085 microprocessor loads Accumulator contents directly from address location 8050H or defining the memory and transferring data by using MOV A,M. This can be done by MOVX A,@DPTR in 8051 microcontroller. In place of SUB and SBB, there is a single instruction used SUBB A, Source; is used for subtraction in 8051 microcontroller.

The stack is worked on the principle of last in first out basis. In 8085, LXI SP,FFFFH instruction loads SP register with FFFFH. PUSH B, PUSH D instructions will push BC and DE pairs into stack and the SP location will be decremented by two in each PUSH instruction. Similarly POP D and POP B are used to retrieve the registers contents. In each POP instruction the SP value is incremented by two as shown in figure 3a.

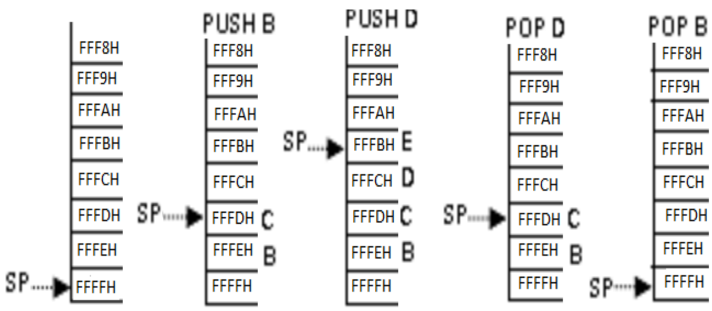

Fig. 3a. Stack operation in 8085 Microprocessor

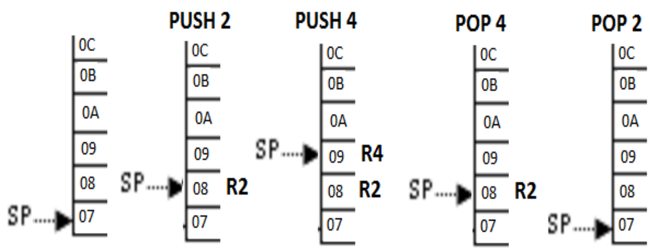

Fig. 3b. Stack operation in 8051 microcontroller

In 8051, the default location of SP is 07. By pushing R2 & R4 registers into stack use PUSH 2 and PUSH 4 instructions and they will be retrieved by using POP R4 & POP R2 instructions. Each push will increment SP by 1 and each POP will decrement by 1 as shown in figure 3b. If bank 1 registers are used in the program, SP should be reloaded with a new location in scratch pad memory. For example to load SP with 40H use instruction MOV SP, #40H.In all 8 bit arithmetic and logical operations of both the 8085 microprocessor and 8051 microcontroller one of the operand should in Accumulator and the destination register is Accumulator only. But for 16 bit addition in 8085, one of the operand is the contents of HL pair and the destination is the contents of HL pair only. For example in DAD D instruction, (HL) = (HL) + (DE). All jump instructions in 8085 can able to jump anywhere 0000 –FFFFH. But, in 8051, the jump instructions can be classified into Long and short jumps. All long jump instructions are 3 byte instructions but all short jumps are two byte instructions only to save the program memory. Similarly the LCALL is a 3 byte instruction and ACALL is a two byte instruction only.

Table 2. Similarities and the differences in assembly language instructions of 8085 and 8051

S.No.	Assembly language instructions of 8085	Assembly language instructions of 8051
1	**Data Transfer operations:** The MVI instruction is used to load the data in to a register. Ex: MVI A,05 ; (A) = 05. MOV instruction is used to transfer either from memory to register and vice versa or from register to register. IN & OUT instructions are used to transfer the data through ports. LXI is used to load a register pair, STAX & STA are the instructions used to store the Accumulator contents in a memory. No flags are affected with these operations.	MOV Destination,Source ; Destination = Source, Source contents does not change. Ex 1: MOV A,#55H ; (A) = 55H. The ' # ' signifies that it is a value otherwise it is a memory address. Ex 2: MOV R5,#0F9H ; (R5) = F9H. The 0 between # and F to indicate that F is a hex number and not a letter. The size of destination & Source should be equal. Ex 3: MOV DPTR, #2040H ; (DPTR) = 2040H, which can also be entered as MOV DPH,#20H & MOV DPL,#40H.
2	**Arithmetic operations:** ADD, ADI, ADC, ACI, DAD, SUB, SUI, SBB are for addition and subtraction, in which one of the data and destination is Accumulator. The INR, INX, DCR, DCX are for incrementing and decrement	(i) ADD A,Source ; (A) = (A) + Source, (ii) ADDC A,Source ; (A) = (A) + (Source) + CY, (iii) DA A for decimal adjustment. (iv) SUBB A, Source ; (A) = (A) – (Source) – CY (v) MUL AB ; (A) = (A) x (B), Upper byte

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

	operations. Flags are affected. DA A is for decimal adjustment of A. Multiplication can be in repeated addition and division can be done in repeated subtraction process.	of product in (B). (vi) DIV AB ; (A) = (A) / (B), the Quotient is in A and Remainder is in B register. INC, DEC are for increment and decrement operations.
3	**Logic Operations:** ANA, ANI, ORA, ORI, XRA, XRI and CMA instructions are used to perform bit wise logical operations. All these instructions are in relation to the contents of Accumulator and no flags are affected	ANL Dest,Source ; (Dest) = (Dest) AND (Source), ORL Dest,Source ; (Dest) = (Dest) OR (Source), XOR Dest,Source ; (Dest) = (Dest) XOR (Source), CPL A ; (A) =($\bar{A}$)
4	**Branch Operations:** Various types of Jump (Un conditional and conditional), Call and Restart instructions are used to change the execution sequence of the program. All types of Jump and Call instructions are 3 byte instructions.	Two types of jumps, LJMP (3 byte) & SJMP (2 byte). All conditional jumps are short jumps only. Two types of call LCALL (3 byte) to call subroutine anywhere in the memory and ACALL (2 byte) to call a routine within the page.
5	**Machine Control Operations:** HLT, NOP and Interrupt instructions are used to indicate the end of the program, no operation and interrupting the processor through external interrupts.	ORG for origin, END for last line of a program, EQU for counter constant. The Define byte (DB) is used to define data. It defines ASCII strings larger than two characters. Ex: DB 0011 0101B binary, DB 39H Hex, DB “2591” ASCII.
6	**Logic Operation Rotate:** RLC &RAL are used to rotate Accumulator contents left without and with carry; RAC & RAR are used to rotate Accumulator contents right without and with carry. All are one byte instructions only. XCHG exchanges HL pair contents with DE pair contents.	RR A ; Rotate right the Accumulator contents. RL A ; Rotate left the Accumulator contents. RRC A ; Rotate right Accumulator contents through CY. RLC A ; Rotate left Accumulator contents through CY. SWAP A ; Swapping Accumulator bytes.
7	**Logic Operation Compare:** CMP & CPI ;Compare Accumulator contents with register / with immediate data by subtracting the data from A. If A < R / M; Cy = 1 & Z = 0; If A = R / M; Cy = 0 & Z = 1,If A > R / M; Cy = 0 & Z = 0	CJNE dest,Source,Relative address ; Compare destination contents with Source contents and jump if not equal. Ex: CJNE A,#55,Next ; If (A) $\neq$ 55H, then jump to Next. Even though it is a conditional jump, It is a 3 byte instruction. No need of involving A in compare instruction
8	**Word Size:** All instructions are classified in 1, 2 and 3 byte instructions based on whether it has only opcode or opcode followed by data or opcode followed by address.	All instructions classified in 1, 2 and 3 byte instructions based on whether it has only opcode or opcode followed by data or opcode followed relative address or full address.
9	**CY flag bit related instructions:** CMC ;Compliment CY, STC ; Set the Carry status, JC target ; jump to the target if CY = 1, JNC target ; jump to the target if CY≠ 1 CC target ; Call subroutine labelled by target if CY = 1 CNC target ; Call subroutine labelled by target if CY≠ 1	SETB C; CLR C; CPL C; MOV b,C; MOV c,b; JNC target; JC target; target is the relative address. ANL c,bit ; AND CY with bit and save it on CY ANL c,/bit ; AND CY with inverted bit and save it on CY ORL c,bit ; OR CY with bit and save it on CY ORL c,/bit ; OR CY with inverted bit and save it on CY
10	**Zero flag bit related instructions:** JZ target; jump to the target if Z = 1, target is 16 bit address. JNZ target ; jump to the target if Z≠ 1, CZ target ; Call subroutine if Z = 1 CNZ target ; Call subroutine labelled by target if Z≠ 1	Instead of zero flag bit Accumulator contents are used. JZ ; Jump if A = 0, JNZ ; Jump if A $\neq$ 0 DJNZ ; decrement and jump if A $\neq$ 0 CJNE A,byte ; jump if A = byte.

THE SIMILARITIES AND DIFFERENCES IN ADDRESSING MODES OF 8085 AND 8051: The programmer must know the source and the destination of the data before and after execution of the instruction. The various ways of accessing data are called addressing modes. The similarities and differences in addressing modes of 8085 microprocessor and 8051 microcontroller were given in table 3. The SFRs have addresses between 80H –FFH, not all the addresses are used by SFRs. The locations 80H –FFH are reserved and must not be used by the programmer. The SFRs can be accessed by their names or by their addresses. In the absence of @ sign in register indirect addressing mode, MOV will be interpreted as an instruction moving the contents of register R0 to A, instead of the contents of memory location pointed by R0. Looping is not possible in direct addressing mode, but it is possible in register indirect addressing mode.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

Table 3. The similarities and differences in addressing modes of 8085 and 8051

S. No.	Addressing modes of 8085 microprocessors	Addressing modes of 8051 microcontroller
1	**Immediate Addressing Mode:** The operand is specified in within the instruction itself. Ex: 1. MVI A,05 instruction loads A with 05 data, 2. ADI 06 instruction adds accumulator contents with 06 and save the result in accumulator. 3. LXI H,8050H loads (HL) = 8050H.	The operand comes immediately after the opcode. The immediate data must be preceded by the pound sign '#'. Ex. 1. MOV A,#25H ; (A) = 25H 2. MOV DPTR,#8050H ; (DPTR) = 8050H 3. MOV P1,#55H ; Transfer 55H to port 1. Source and destination register must be match in size.
2	**Register Addressing Mode:** The operand is in one of the general purpose registers, in accumulator or in memory. The opcode specifies the address of the registers in addition to the operation to be performed. Ex. 1. MOV A,B; Transfer (B) to (A). The opcode 78 specifies this operation. 2. ADD B; Add (A) with (B) and save the result in (A). The opcode 80 specifies this operation.	It involves the use of registers to hold the data to be manipulated. Ex. 1. MOV A, R0 ; Transfer (R)) to (A) 2. MOV R2,A ; Transfer (A) to (R2) 3. MOV R5,R6; is an invalid instruction, if we want to transfer (R6) to (R5), first transfer (R6) to (A) and then transfer (A) to (R5). Source register and destination register must be match in size.
3	**Direct Addressing Mode:** The address of the operand is given in the instruction itself. Ex: 1. STA 8050H; Store (A) in a memory location 8050H, in which the three bytes 32H,50H,80H specifies opcode, lower and higher bytes of the address. 2. OUT 30H; Transfer (A) to port A address. The D3, 30H indicate the code and port A address. 3. IN 31H; Transfer port B contents to (A). The DB, 31H indicate the code and port B address.	The data's at 128 bytes of on chip RAM (00 – 7FH) and 128 bytes external RAM (80H – FFH) can be accessed by direct addressing mode. Ex. 1. MOV R0,40H ; (R0) = contents of 40H = (40H) 2. MOV A,50H ; (A) = contents of 50H = (50H) 3. MOV A,4 = MOV A,R4 4. MOV 0E0H,#55H = MOV A,#55H Only direct addressing mode is allowed for pushing on to the stack and it must be used for POP instruction as well.
4	**Register Indirect Addressing Mode:** The address of the operand is specified by a register pair Ex. 1. LXI H, 8060H; (HL) = 8060H = (M) MOV A,M; Move the contents of memory location, whose address is in HL pair to the Accumulator. 2. ADD M; (A) = (A) + (M) = (A) + (HL)	A register is used as a pointer to the data. Only R0 & R1 are used for this purpose. When R0 & R1 hold the address of RAM locations, they must be preceded by the '@' sign. Otherwise register contents will be transferred. Ex. 1. MOV A,@R0 ; (A) = contents of R0 = (R0) 2. MOV @R1,A ; contents of R1 = (R1) = (A)
5	**Implicit Addressing Mode:** The instructions do not require the address of the operand. These instructions are meant for Accumulator only. Ex. 1. CMA; Compliment the Accumulator contents 2. RAL; Rotate contents of (A) to left with carry. 3. RAR; Rotate contents of (A) to left with carry.	**Indexed addressing mode:** It is widely used in accessing data elements of look up table entries located in the program ROM space of the 8051. The instruction used for this purpose is MOVC A,@A+DPTR. Because the data elements are stored in the program (Code) space, it uses MOVC instead of MOV. The 'C' means code.

SIMILARITIES AND DIFFERENCES IN ASSEMBLY LANGUAGE PROGRAMS OF 8085 AND 8051: The assembly language program of 8085 microprocessor is appeared in five field format [7]. The address field, opcode field, Label field, mnemonic field and the comments/ description field. But the assembly language program of 8051 can be appeared as four field format only. The Assembly language program for 16 bit addition and multiplication of two hexadecimal numbers for both 8085 and 8051 were given Table 4.

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

Table 4. The similarities and differences in assembly language programs of 8085 and 8051

Assembly language programs of 8085 microprocessors

1. To write and execute an ALP of 8085 microprocessor to add two 16 bit data, stored from the memory locations 8050H using Double Addition (DAD) instruction. The program is given below:

Address	Opcode	Lbl	Mnemonic	Comments
8000H	2A,50,80		LHLD 8050H	(HL) = (8050H)
8003H	EB		XCHG	(DE) ↔ (HL)
8004H	2A,52,80		LHLD 8052H	(HL) = (8052H)
8007H	0E,00		MVI C,00	(C) = 00,
8009H	19		DAD D	(HL)=(HL) + (DE)
800AH	D2,0E,80		JNC NEXT	If CY=0, to nxt
800DH	0C		INR C	If CY=1,store in C
800EH	22,0E,80		SHLD 8054H	(HL) stored in 8054H & 8055H
8011H	79	nxt	MOV A,C	(A) = (C) = CY
8012H	32,56,80		STA 8056H	(A)=(8056H)= CY
8015H	76		HLT	Halt the program

In the 8085 program, the first 16 bit data is stored in 8050H & 8051H and second 16 bit data is stored in 8052H & 8053H locations.
In the 8051 program, LB stands for Lower byte and HB stands for Higher byte. The lower bytes of two 16 bit data are stored in 50H and 51H and higher bytes are stored in 52H and 53H. If the instruction MOV @R0,RX is not working, first transfer RX to A and then transfer to the RAM memory and vice versa.

Assembly language programs of 8051 microcontroller

1. To write and execute an ALP of 8051 to add two 16 bit data, stored from the RAM memory locations 50H and store the result in next memory. The program is given below:
ORG 8000 ; origin address of the program

Addr	Lbl	Mnemonic	Comments
8000H		MOV R3,#0	;(R3) = 00
8002H		MOV R0,#50H	;(R0)=50H=LB of D1
8004H		MOV A,@R0	; (A)= LB of D1
8005H		INC R0	;(R0)=51H= LB of D2
8006H		ADD A,@R0	; (A) = (A) + (51H)
8007H		MOV R2,A	; (R2) = Sum of LBs
8008H		INC R0	;(R0)=52H=HB of D1
8009H		MOV A,@R0	; (A)= HB of Data1
800AH		INC R0	;(R0)=53H=HB of D2
800BH		ADDC A,@R0	;(A)=(A) + (53H) + CY
800CH		JNC NXT	;If CY=0, jump to nxt
800EH		INC R3	; If CY=1, store in R3
800FH	nxt:	INC R0	;(R0) = 54H
8010H		MOV @R0,R2	; (54H) = Sum of LBs
8011H		INC R0	;(R0) = 55H
8012H		MOV @R0,A	; (54H) = Sum of HBs
8013H		INC R0	;(R0) = 56H
8014H		MOV @R0,R3	; (56H) = CY condition
8015H		LCALL 0003	; End of the program

8085: 2. To write and execute an ALP of 8085 microprocessor to multiply two 8 bit data's stored in 8050H the program is given below:

Address	Op code	Label	Mnemonic	Comments
8000H			LXI H,8050H	(HL)=(8050H)
8003H			MVI B,00	(B) = 00, for CY
8005H			MOV A,B	(A) = 00
8006H			MOV C,M	(C) = Data 1
8007H			INX H	(HL)=(8051H)
8008H			MOV D,M	(D) = Data 2
8009H		BCK:	ADD D	(A) = (A) + (D)
800AH			JNC NXT	If CY=0, to NXT
800DH			INR B	If CY=1,store in B
800EH		NXT:	DCR C	Decrement C
800FH			JNZ BCK	If C≠0, to BCK
8012H			INX H	(HL) = 8051H
8013H			MOV M,A	(8051H) = Product
8014H			INX H	(HL) = 8052H
8015H			MOV M,B	(8052H) = CY
8016H			HLT	Halt the program

8051: 2. To write and execute an ALP of 8051 to multiply two 8 bit data's stored in 50H the program is given below:

Address	Label	Mnemonic	Comments
8000H		MOV R0,#50H	;(R0)=50H= Data1
8002H		MOV A,@R0	; (A)= Data1
8003H		INC R0	;(R0)=51H= Data2
8004H		MOV 0F0,@R0	; (B)= Data2
8006H		MUL AB	; (A) = (A) x (B)
8007H		INC R0	;(R0)=52H
8008H		MOV @R0,A	;(52H) = LB of product
8009H		INC R0	;(R0)=53H
800AH		MOV @R0,0F0	;(53H) = HB of product
800CH		LCALL 0003	; End of the program

Because of MUL AB instruction, the execution of multiplication program is easier with 8051 microcontroller than 8085 microprocessor. It saves the memory space of the central processing unit

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 4, Issue 4, April 2016

IV. CONCLUSIONS

1. The 8085 is the most useful 8 bit general purpose microprocessor and the 8051 is the most familiar and useful 8 bit microcontroller used for controlling applications.
2. A powerful feature of 8051 I/O ports is their capability to access individual bits of the port without altering the rest of the bits in that port. Many special functional registers have the capability to access individual bits without altering other bits in the register.
3. Compare with 8085 microprocessor, 8051 microcontroller has some additional instructions used for setting and resetting the individual bits of single bit accessible registers like A, PSW, TCON, TMOD, SCON etc.;
4. The SP register is an 8 bit register in 8051 microcontroller and direct addressing mode is used to push / pop the contents in stack memory.
5. All conditional jump and all conditional call instructions of 8051 are 2 byte instructions, but all types of jump and all call instructions of 8085 are 3 byte instructions only.

V. ACKNOWLEDGEMENTS

The First author wish to express her thanks to the Principal, Govt. City College, Hyderabad to carry out the above work in Department of Mathematics and the Second author wish to express his sincere thanks to the Former Chairman Prof. Sudershan Singh garu and the present Chairman, Secretary & Correspondent Dr. S. Jai Kishan garu of Bhavan's New Science College for their encouragement and support to undertake the present work and providing the facilities in Department of Physics to write and execute the assembly language programs in M.Sc. Applied Electronics Laboratories of Department of Physics. He also expresses his thanks to his colleagues in Dept. of Physics and M.Sc. Applied Electronic students for their support.

REFERENCES

1. Ramesh S. Gaonkar; Microprocessor Architecture, Programming and Applications with the 8085 / 8080A, Second Edition, publishing for one world New Age International Pvt. Ltd, Publishers.
2. B. Ram and Snjay Kumar; Fundamentals of Microprocessors and microcontrollers, Seventh revised edition, Dhanpat Rai Publications (P) Ltd., New Delhi-110002
3. Muhammad Ali Mazidi, Jainice Gillispie Mazidi and Rolin D. McKinlay;
The 8051 Microcontrollers and Embedded Systems using Assembly and C, Second Edition, Eastern Economy Edition-2006, Prentice Hall of India Private Limited, New Delhi-110001.
4. Dr. D. Chinni Krishna, U. Sridevi and B. Anjaiah
International Journal of Engineering Science and Computing (IJESC), Volume 6, Issue No. 4, April 2016, Pages 3523 – 3528.
5. MPS 85-3 User's Manual by Electro systems Associates Pvt Ltd., Bangalore-560021, Karnataka, India
6. ALS-SDA-8051-MEL User's Manual, Advanced Electronics, Banagalore-560058, Karnataka, India.
7. Dr. D. Chinni Krishna and Prof. G. Pushpa Chakrapani
M. Sc. Physics, Laboratory Manual & Record on Memory Devices & Microprocessors of Dr.B.R. Ambedkar Open University, Hyderabad.

BIOGRAPHY

U . Sridevi is a Lecturer & Head, Department of Mathematics, Govt. City College, Nayapul, Hyderabad, Telangana State, India. She received Master of Science (M.Sc.) degree in Mathematics in 1998 from Osmania University, Hyderabad, Telangana State, India. She received Master of Philosophy (M.Phil.) degree in Mathematics in 2004 from Madurai Kamaraj University, Tamilnadu, India. Her research interests are Numerical Computing and Writing and executing computer programs.

Dr. D. Chinni Krishna is a Senior Lecturer in Department of Physics, Bhavan's New Science College (PG), Narayanaguda, Hyderabad, Telangana State, India. His Associate Professorship is due since 2013. He received Master of Science (M.Sc.) (Technology) in Applied Electronics with distinction in 1997 from Osmania University, Hyderabad, Telangana State, India. He received Post graduate diploma in Computer Science (PGDCS) in 2003 from Hyderabad Central University, Hyderabad, Telangana State, India. He received Doctor of Philosophy (Ph.D.) in Physics in 2010 from Osmania University, Hyderabad, Telangana State, India. His research interests are Condensed Matter Physics, Nanomaterials, Digital Electronics Circuits, Microprocessors and Microcontrollers.