***MADHAV INSTITUTE OF TECHNOLOGY  AND SCIENCE, GWALIOR (M.P.) -474005***

(A Govt. Aided UGC Autonomous & NAAC Accredited Institute Affiliated to RGPV, Bhopal)



DEPARTMENT OF CSE and IT

PRACTICLE FILE

MICROPROCESSOR AND INTERFACING

160504

SUBMITTED TO:

PUNIT KUMAR JOHARI

Assistant Professor IT3rd YEAR(5THsem)

SUBMITTED BY:

Shivansh Shrivastava

0901IT181056

Session July-November, 2020

Content

|  |  |  |  |
| --- | --- | --- | --- |
| S.No. | Experiment name | Date of Practical | Signature of Teacher |
| 1 | Write an assembly language program to perform the subtraction of two 8-bitnumber using 8085/8086 instruction set |  |  |
| 2 | Write an assembly language program to move data block starting at location ‘X’ to location ‘Y’ without overlap using 8085/8086 instruction set. |  |  |
| 3 | Write an assembly language program to move data block starting at location ‘X’ to location ‘Y’ with overlap using 8085/8086 instruction set. |  |  |
| 4 | Write an assembly language program to arrange set of 8-bit numbers starting at location in ASCENDING/DESCENDING order. Display the stored vector in address data field using 8085/8086 instruction set. |  |  |
| 5 | 1. Write an assembly language program to perform the multiplication of two 8-bit numbers using 8085/8086 instruction set. |  |  |
| 6 | 1. Write an assembly language program to perform the division of two 8-bit numbers using 8085/8086 instruction set. |  |  |
| 7 | 1. Write an assembly language program to find the larger number in array of data using 8085/8086 instruction set. |  |  |
| 8 |  |  |  |
| 9 | Write an assembly language program to convert given hexadecimal number into its equivalent BCD number using 8085/8086 instruction set. |  |  |
| 10 | Write an assembly language program to convert given hexadecimal number into its equivalent BCD number using 8085/8086 instruction set. |  |  |
| 11 | Write an assembly language program to convert given ASCII character into its equivalent hexadecimal number using 8085/8086 instruction set. |  |  |

**Experiment No. 1**

**AIM:** Write an assembly language program to perform the subtraction of two 8-bitnumber using 8085/8086 instruction set.

**Instruments Required:** 1. 8085/8086 Microprocessor Kit

2. +5V Power supply

**Theory**: In microprocessor 8085, the Sub instruction used 2s complement method for subtraction.Whwn the firest operand is larger ,the result will be positive.It will not enable the carry flag after completing the subtraction.

**Algorithm**

1. Load 00 in a register C (for borrow)
2. Load two 8-bit number from memory into registers
3. Move one number to accumulator
4. Subtract the second number with accumulator
5. If borrow is not equal to 1, go to step 7
6. Increment register for borrow by 1
7. Store accumulator content in memory
8. Move content of register into accumulator
9. Store content of accumulator in other memory location
10. Stop

**PROGRAM**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Labels*** | ***Memory***  ***Address*** | ***Machine Codes/ OPCODE*** | ***Mnemonics*** | ***Comments*** |
| START | 0000 | 0E | MVI C,00 |  |
|  | 0002 | 2A | LHLD 2500 |  |
|  | 0005 | 7C | MOV A,H |  |
|  | 0006 | 95 | SUB L |  |
|  | 0007 | D2 | JNC 200B |  |
|  | 000A | 0C | INR C |  |
|  | 000B | 32 | STA 2502 |  |
|  | 000E | 79 | MOV A,C |  |
|  | 000F | 32 | STA 2503 |  |
|  | 0012 | 76 | HLT |  |
| END | 2503 | 01 |  |  |

**Experimental Results**

|  |  |  |  |
| --- | --- | --- | --- |
| **Input DATA** |  | **RESULT** |  |
| Memory location | Data | Memory location | Data |
| 2500 | **03** | 2502 | **01** |
| 2501 | **04** | 2503 | 01 |
|  |  |  |  |

**Conclusion**: Thus the program to subtract two 8-bit number was executed

**Precautions:**

1. Properly connect the 8085 microprocessor kit with power supply terminals.
2. Switch on the power supply after checking connections
3. Handle the Trainer kit carefully.

**Experiment No. 2:**

**AIM:-** Write an assembly language program to move data block starting at location ‘X’ to location ‘Y’ without overlap using 8085/8086 instruction set.

**Instruments Required**: 1. 8085/8086 Microprocessor Kit

2. +5V Power supply

**Theory:** The non-overlapping block movement is relatively an easy task. Here the block is starting at position X, we have to move it to position Y. The location Y is far away from X. So Y > X + block size.In this program, the data are stored at location 2000H, and we will place it 2010H. The block size is stored at location 2000H.We are loading the source address into HL pair, and destination address into DE register pair. We are simply taking the data from the source and placing it at the destination.

**Algorithm:**

1. Store the starting address of both blocks in 2 different register pairs
2. Now exchange the contents at the addresses in both register pairs
3. Increment the values of both register pairs
4. Decrements count by 1
5. If count is not equal to 0 repeat steps 3 to 5

**PROGRAM:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Labels*** | ***Memory***  ***Address*** | ***Machine Codes/ OPCODE*** | ***Mnemonics*** | ***Comments*** |
| PROG | 2000 | 16 | MVI D, 06 |  |
|  | 2001 | 06 |  |  |
|  | 2002 | 21 | LXI H,F100 |  |
|  | 2003 | 00 |  |  |
|  | 2004 | F1 |  |  |
|  | 2005 | 01 | LXI H,B200 |  |
|  | 2006 | 00 |  |  |
|  | 2007 | F2 |  |  |
| UP | 2008 | 7E | MOV A,M |  |
|  | 2009 | 02 | STAX B |  |
|  | 200A | 23 | INX H |  |
|  | 200B | 03 | INX B |  |
|  | 200C | 15 | DCR D |  |
|  | 200D | C2 | JNZ UP |  |
|  | 200E | 08 |  |  |
|  | 200F | 20 |  |  |
| END | 2010 | 2F | RST 1 |  |

**Experimental Results:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Input DATA** |  | **RESULT** |  |
| Memory location | Data | Memory location | Data |
| F100 | **AA** | F200 | **AA** |
| F101 | **BB** | F201 | BB |

**Conclusion:** Thus the program to move data without overlap was executed

**Precautions:**

1. Properly connect the 8085 microprocessor kit with power supply terminals.
2. Switch on the power supply after checking connections
3. Handle the Trainer kit carefully.

**Experiment No. 3:**

**AIM:-** Write an assembly language program to move data block starting at location ‘X’ to location ‘Y’ with overlap using 8085/8086 instruction set.

**Instruments Required**: 1. 8085/8086 Microprocessor Kit

2. +5V Power supply

**ALGORITRHM:**

1. The overlapping block movement is tricky; we need to use some special tricks to move a block in overlapping address locations. Here the block is starting at position X, we have to move it to position Y.
2. The location Y is inside the block. So Y < X + block size.
3. In this program, the data are stored at location 8010H, and we will place it 8014H. The block size is stored at location 8000H.
4. To move a block into some overlapping locations, we will start from the end element of the block. So at first the last element of the block is moved, then the previous bytes are moved.

**PROGRAM:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ADDRESS | LABELS | MNEMONICS | HEXCODE | COMMENTS |
| F000 |  | LXI H,8000 | 21,00,80 |  |
| F001 |  |  |  |  |
| F002 |  |  |  |  |
| F003 |  | MOV C,M | 4E |  |
| F004 |  | MVI B,00 | 06 |  |
| F005 |  |  | 00 |  |
| F006 |  | LXI H,8010 | 21 |  |
| F007 |  |  | 10 |  |
| F008 |  |  | 80 |  |
| F009 |  | LXI D,8014 | 11 |  |
| F00A |  |  | 14 |  |
| F00B |  |  | 80 |  |
| F00C |  | DAD B | 09 |  |
| F00D |  | DCX H | 2B |  |
| F00E |  | XCHG | EB |  |
| F00F |  | DAD B | 09 |  |
| F010 |  | DCX H | 2B |  |
| F011 |  | XCHG | EB |  |
| F012 | LOOP | MOV A,M | 7E |  |
| F013 |  | STAX D | 12 |  |
| F014 |  | DCX D | 1B |  |
| F015 |  | DCX H | 2B |  |
| F016 |  | DCR C | 0D |  |
| F017 |  | JNZ LOOP | C2 |  |
| F018 |  |  | 12 |  |
| F019 |  |  | F0 |  |
| F01A |  | HLT | 76 |  |

**Experimental Results**

|  |  |  |  |
| --- | --- | --- | --- |
| **Input DATA** |  | **RESULT** |  |
| Memory location | Data | Memory location | Data |
| 8000 | **05** | 8000 | **05** |
| 8014 | **15** | 8018 | 15 |

**Conclusion:**

So the program to move data with overlapping was executed.

**Precautions:**

1. Properly connect the 8085 microprocessor kit with power supply terminals.
2. Switch on the power supply after checking connections
3. Handle the Trainer kit carefully.

**Experiment No. 4:**

**AIM:-**Write an assembly language program to arrange set of 8-bit numbers starting at location in ASCENDING/DESCENDING order. Display the stored vector in address data field using 8085/8086 instruction set.

**Instruments Required**: 1. 8085/8086 Microprocessor Kit

2. +5V Power supply

**Theory:** An 8 bit data is arrange in sorted form or in ascending order by using various assembly language and registers. Set the memory location at 2000H

**ALGORITHM:**

1. Load data from offset 500 to register CL (for count).
2. Travel from starting memory location to last and compare two numbers if first number is greater than second number then swap them.
3. First pass fix the position for last number.
4. Decrease the count by 1.
5. Again travel from starting memory location to (last-1, by help of count) and compare two numbers if first number is greater than second number then swap them.
6. Second pass fix the position for last two numbers.
7. Repeate.

**PROGRAM:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
| Memory  Address | ***Labels*** | ***Mnemonics*** | ***Machine Codes/ OPCODE*** | ***Comments*** |
| 2000 |  | LDA F100 | 3A |  |
| 2001 |  |  | 00 |  |
| 2002 |  |  | F1 |  |
| 2003 |  | DCR A | 3D |  |
| 2004 |  | MOV C,A | 4F |  |
| 2005 |  | MOV B,C | 41 |  |
| 2006 |  | LXI H,F200 | 21 |  |
| 2007 |  |  | 00 |  |
| 2008 |  |  | F2 |  |
| 2009 | UP | MOV A,M | 7E |  |
| 200A |  | INX H | 23 |  |
| 200B |  | CMP M | BE |  |
| 200C |  | JC DOWN | DA |  |
| 200D |  |  | 14 |  |
| 200E |  |  | 20 |  |
| 200F |  | MOV D,M | 56 |  |
| 2010 |  | MOV M,A | 77 |  |
| 2011 |  | DCX H | 2B |  |
| 2012 |  | MOV M,D | 72 |  |
| 2013 |  | INX H | 23 |  |
| 2014 | DOWN | DCR B | 05 |  |
| 2015 |  | JNZ UP | C2 |  |
| 2016 |  |  | 09 |  |
| 2017 |  |  | 20 |  |
| 2018 |  | DCR C | 0D |  |
| 2019 |  | JNZ 2005 | C2 |  |
| 201A |  |  | 05 |  |
| 201B |  |  | 20 |  |
| 201C | RESULT | RST 1 | CF |  |
|  |  |  |  |  |

**Experimental Results :**

At memory location F100 we take count of 4.

|  |  |  |  |
| --- | --- | --- | --- |
| **Input DATA** |  | **RESULT** |  |
| Memory location | Data | Memory location | Data |
| F200 | **DD** | F200 | AA |
| F201 | **CC** | F201 | BB |
| F202 | **BB** | F202 | CC |
| F203 | **AA** | F203 | DD |

**Conclusion:**

We successfully arrange data in ascending order .

**Precautions:**

1. Properly connect the 8085 microprocessor kit with power supply terminals.
2. Switch on the power supply after checking connections
3. Handle the Trainer kit carefully.

**EXPERIMENT 5:-**

**AIM:** Write an assembly language program to perform the multiplication of two 8-bit numbers using 8085/8086 instruction set

**Instruments Required**: 1. 8085/8086 Microprocessor Kit

2. +5V Power supply

**Theory:**

Here we study the multiplication of 8 bit number .

**ALOGRITHM:**

1) Start the program by loading HL register pair with address of memory location.

2) Move the data to a register (B register).

3) Get the second data and load into Accumulator.

4) Add the two register contents.

5) Check for carry.

6) Increment the value of carry.

7) Check whether repeated addition is over and store the value of product and carry

in memory location.

8) Terminate the program.

PROGRAM:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MEMORY  ADDRESS | LABELS | MNEMONICS | OPCODE | COMMENTS |
| 2000 |  | LHLD 2501 | 2A |  |
| 2001 |  |  | 01 |  |
| 2002 |  |  | 25 |  |
| 2003 |  | XCHG | EB |  |
| 2004 |  | LDA 2503 | 3A |  |
| 2005 |  |  | 03 |  |
| 2006 |  |  | 25 |  |
| 2007 |  | LXI H,0000 | 21 |  |
| 2008 |  |  | 00 |  |
| 2009 |  |  | 00 |  |
| 200A |  | MVI C,08 | 0E |  |
| 200B |  |  | 08 |  |
| 200C | LOOP | DAD H | 29 |  |
| 200D |  | RAL | 17 |  |
| 200E |  | JNC AHEAD | D2 |  |
| 200F |  |  | 12 |  |
| 2010 |  |  | 20 |  |
| 2011 |  | DAD D | 19 |  |
| 2012 | AHEAD | DCR C | 0D |  |
| 2013 |  | JNZ LOOP | C2 |  |
| 2014 |  |  | 0C |  |
| 2015 |  |  | 20 |  |
| 2016 |  | SHLD 2504 | 22 |  |
| 2017 |  |  | 04 |  |
| 2018 |  |  | 25 |  |
| 2019 |  | HLT | 76 |  |

**Experimental Results :**

**At memory location F100 we take count of 4.**

|  |  |  |  |
| --- | --- | --- | --- |
| **Input DATA** |  | **RESULT** |  |
| Memory location | Data | Memory location | Data |
| 2001 | **84H** | 2004 | 58 |
| 2003 | **56H** | 2005 | 2C |

**Conclusion:**

We successfully studied the multpliction of number.

**Precautions:**

1. Properly connect the 8085 microprocessor kit with power supply terminals.

1. Switch on the power supply after checking connections
2. Handle the Trainer kit carefully.

**EXPERIMENT 6**

**AIM:** Write an assembly language program to perform the division of two 8-bit numbers using 8085/8086 instruction set.

**Instruments Required**: 1. 8085/8086 Microprocessor Kit

2. +5V Power supply

**ALGORITHM:**

1) Start the program by loading HL register pair with address of memory location.

2) Move the data to a register(B register).

3) Get the second data and load into Accumulator.

4) Compare the two numbers to check for carry.

5) Subtract the two numbers.

6) Increment the value of carry .

7) Check whether repeated subtraction is over and store the value of product and

carry in memory location.

8) Terminate the program.

**PROGRAM:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MEMORY  ADRESS | LABELS | MNEMONICS | OPCODE | COMMENTS |
| 2000 |  | LHLD 2501 | 2A |  |
| 2001 |  |  | 01 |  |
| 2002 |  |  | 25 |  |
| 2003 |  | LDA 2503 | 3A |  |
| 2004 |  |  | 03 |  |
| 2005 |  |  | 25 |  |
| 2006 |  | MOV B,A | 47 |  |
| 2007 |  | MVI C,08 | 0E |  |
| 2008 |  |  | 08 |  |
| 2009 | LOOP | DAD H | 29 |  |
| 200A |  | MOV A,H | 7C |  |
| 200B |  | SUB B | 90 |  |
| 200C |  | JC AHEAD | DA |  |
| 200D |  |  | 11 |  |
| 200E |  |  | 20 |  |
| 200F |  | MOV H,A | 67 |  |
| 2010 |  | INR L | 2C |  |
| 2011 | AHEAD | DCR C | 0D |  |
| 2012 |  | JNZ LOOP | C2 |  |
| 2013 |  |  | 09 |  |
| 2014 |  |  | 20 |  |
| 2015 |  | SHLD 2504 | 22 |  |
| 2016 |  |  | 04 |  |
| 2017 |  |  | 25 |  |
| 2018 |  | HLT | 76 |  |

**Experimental Results :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Input DATA** |  | **RESULT** |  |
| Memory location | Data | Memory location | Data |
| 2001 | **9BH** | 2004 | F2H |
| 2002 | **48H** | 2005 | 07H |
| 2003 | **1AH** |  |  |

**Conclusion:**

We successfully studied the Division of number.

**Precautions:**

* Properly connect the 8085 microprocessor kit with power supply terminals.
* Switch on the power supply after checking connections
* Handle the Trainer kit carefully.
* **EXPERIMENT 7**

**AIM:-** Write an assembly language program to find the larger number in array of data using 8085/8086 instruction set.

**Instruments Required**: 1. 8085/8086 Microprocessor Kit

2. +5V Power supply

**Algorithm –**

1. We are taking first element of array in A
2. Comparing A with other elements of array, if A is smaller then store that element in A otherwise compare with next element
3. The value of A is the answer

­­­­­­­­­­­­

**PROGRAM:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MEMORY  ADDRESS | LABEL | MNEMONICS | OPCODE | COMMENTS |
| 2000 |  | LXI H,2050 | 21 |  |
| 2001 |  |  | 50 |  |
| 2002 |  |  | 20 |  |
| 2003 |  | MOV C,M | 4E |  |
| 2004 |  | DCR C | 0D |  |
| 2005 |  | INX H | 23 |  |
| 2006 |  | MOV A,M | 7E |  |
| 2007 |  | INX H | 23 |  |
| 2008 |  | CMP M | BE |  |
| 2009 |  | JNC 200D | D2 |  |
| 200A |  |  | 0D |  |
| 200B |  |  | 20 |  |
| 200C |  | MOV A,M | 7E |  |
| 200D |  | DCR C | 0D |  |
| 200E |  | JNZ 2007 | C2 |  |
| 200F |  |  | 07 |  |
| 2010 |  |  | 20 |  |
| 2011 |  | STA 3050 | 32 |  |
| 2012 |  |  | 50 |  |
| 2013 |  |  | 30 |  |
| 2014 |  | HLT | 76 |  |

**Experimental Results :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Input DATA** |  | **RESULT** |  |
| Memory location | Data | Memory location | Data |
| 2001 | **04H** | 3050 | 55H |
| 2002 | **1AH** |  |  |
| 2003 | **55H** |  |  |

**Conclusion:**

We successfully find the largest number .

**Precautions:**

1. Properly connect the 8085 microprocessor kit with power supply terminals.
2. Switch on the power supply after checking connections
3. Handle the Trainer kit carefully.

**EXPERIMENT 9**

**AIM:** Write an assembly language program to convert given hexadecimal number into its equivalent BCD number using 8085/8086 instruction set.

**Instruments Required**: 1. 8085/8086 Microprocessor Kit

2. +5V Power supply

**THEORY:-**

Here we convert the the binary number (hexadecimal ) to the BCD format.

Binary-coded decimals are an easy way to represent decimal values, as each digit is represented by its own 4-bit binary sequence which only has 10 different combinations. By comparison, converting real binary representation to decimal requires arithmetic operations like multiplication and addition.

**ALGORITHM:-**

1. Here we are taking a number from the memory, and initializing it as a counter.
2. Now in each step of this counter we are incrementing the number by 1, and adjust the decimal value.
3. By this process we are finding the BCD value of binary number or hexadecimal number.
4. We can use INR instruction to increment the counter in this case but this instruction will not affect carry flag, so for that reason we have used ADI 10H

**PROGRAM:-**

#org F000H

#begin F000H

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MEMORY  ADDRESS | LABELS | MNEMONICS | HEXCODE | COMMENTS |
| F000 |  | LXI H,8000 | 21 |  |
| F001 |  |  | 00 |  |
| F002 |  |  | 80 |  |
| F003 |  | MVI D,00 | 16 |  |
| F004 |  |  | 00 |  |
| F005 |  | XRA A | AF |  |
| F006 |  | MOV C,M | 4E |  |
| F007 | LOOP | ADI 01 | C6 |  |
| F008 |  |  | 01 |  |
| F009 |  | DAA | 27 |  |
| F00A |  | JNC SKIP | D2 |  |
| F00B |  |  | 0E |  |
| F00C |  |  | F0 |  |
| F00D |  | INR D | 14 |  |
| F00E | SKIP | DCR C | 0D |  |
| F00F |  | JNZ LOOP | C2 |  |
| F010 |  |  | 07 |  |
| F011 |  |  | F0 |  |
| F012 |  | MOV L,A | 6F |  |
| F013 |  | MOV H,D | 62 |  |
| F014 |  | SHLD 8050 | 22 |  |
| F015 |  |  | 50 |  |
| F016 |  |  | 80 |  |
| F017 |  | HLT | 76 |  |
|  |  |  |  |  |

**Experimental Results :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Input DATA** |  | **RESULT** |  |
| Memory location | Data | Memory location | Data |
| 8000 | **34H** | 8050 | 52H |
|  |  |  |  |
|  |  |  |  |

**Conclusion:**

Here we successfully convert hexdecimal number to bcd .

**Precautions:**

1. Properly connect the 8085 microprocessor kit with power supply terminals.
2. Switch on the power supply after checking connections
3. Handle the Trainer kit carefully.

**EXPERIMENT 10**

**AIM:** Write an assembly language program to convert given hexadecimal number into its equivalent ASCII number using 8085/8086 instruction set.

**Instruments Required**: 1. 8085/8086 Microprocessor Kit

2. +5V Power supply

**THEORY:-**

The general way of converting ascii number from hexdecimal is

Each hexadecimal digit represents a value from zero to fifteen.

To get the ASCII code for a digit-value less than ten [0–9], just add 48 or 0x30 — which is the ASCII code for the digit “0”. This translates the decimal value into the corresponding ASCII character: 0 becomes ‘0’, 1 becomes ‘1’, . . . , 9 becomes ‘9′.

For digit-values of greater than nine, just add 64 or 0x40 (which is one less than the ASCII code for capital ‘A’ ), if you want to use the letters A, B, C, D, E, and F to represent digits with values of ten, eleven, twelve, thirteen, fourteen, and fifteen.

**Algorithm:**

1. Load the given data in accumulator and move to B register
2. Mask the most significant 4 bits(upper nibble) of the Hexa decimal number in accumulator.
3. Call subroutine to get ASCII of least significant 4 bits.
4. Store it in memory
5. Move B register to accumulator and mask the least significant 4 bits(lower nibble).
6. Rotate the upper and lower nibble position.
7. Call subroutine to get ASCII of upper nibble
8. Store it in memory
9. Terminate the program.

**PROGRAM:**

**#org 2000H**

**#begin 2000H**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MEMORY ADDRESS | LABELS | MNEOMONICS | OPCODE | COMMENTS |
| 2000 |  | LDA 2050 | 3A |  |
| 2001 |  |  | 50 |  |
| 2002 |  |  | 20 |  |
| 2003 |  | MOV B,A | 47 |  |
| 2004 |  | ANI 0F | E6 |  |
| 2005 |  |  | 0F |  |
| 2006 |  | CALL SUB1 | CD |  |
| 2007 |  |  | 1A |  |
| 2008 |  |  | 20 |  |
| 2009 |  | STA 2051 | 32 |  |
| 200A |  |  | 51 |  |
| 200B |  |  | 20 |  |
| 200C |  | MOV A,B | 78 |  |
| 200D |  | ANI F0 | E6 |  |
| 200E |  |  | F0 |  |
| 200F |  | RLC | 07 |  |
| 2010 |  | RLC | 07 |  |
| 2011 |  | RLC | 07 |  |
| 2012 |  | RLC | 07 |  |
| 2013 |  | CALL SUB1 | CD |  |
| 2014 |  |  | 1A |  |
| 2015 |  |  | 20 |  |
| 2016 |  | STA 2052 | 32 |  |
| 2017 |  |  | 52 |  |
| 2018 |  |  | 20 |  |
| 2019 |  | HLT | 76 |  |
| 201A | SUB1 | CPI 0A | FE |  |
| 201B |  |  | 0A |  |
| 201C |  | JC SKIP | DA |  |
| 201D |  |  | 21 |  |
| 201E |  |  | 20 |  |
| 201F |  | ADI 07 | C6 |  |
| 2020 |  |  | 07 |  |
| 2021 | SKIP | ADI 30 | C6 |  |
| 2022 |  |  | 30 |  |
| 2023 |  | RET | C9 |  |

**Experimental Results :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input DATA** |  | **RESULT** |  |  |
| Memory location | Data | Memory location | Data | Comments |
| 2050 | **E4H** | 2051 | 34 | ASCII for E |
|  |  | 2052 | 45 | ASCII for E |
|  |  |  |  |  |

**Conclusion:**

Here we successfully convert hexdecimal number to bcd .

**Precautions:**

1. Properly connect the 8085 microprocessor kit with power supply terminals.
2. Switch on the power supply after checking connections
3. Handle the Trainer kit carefully.

**EXPERIMENT:-11**

**AIM:-** Write an assembly language program to convert given ASCII character into its equivalent hexadecimal number using 8085/8086 instruction set.

**Instruments Required**: 1. 8085/8086 Microprocessor Kit

2. +5V Power supply

**Algorithm –**

1. Input the content of 2050 in accumulator.
2. Subtract 30H from accumulator.
3. Compare the content of accumulator with 0AH.
4. If content of accumulator is less than 0A then goto step 6 else goto step 5.
5. Subtract 07H from accumulator.
6. Store content of accumulator to memory location 3050.
7. Terminate the program.

**PROGRAM:-**

**#org 2000H**

**#begin 2000H**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| MEMORY ADDRESS | LABELS | MNEMONICS | OPCODE | COMMENTS |
| 2000 |  | LDA 2050 | 3A |  |
| 2001 |  |  | 50 |  |
| 2002 |  |  | 20 |  |
| 2003 |  | SUI 30 | D6 |  |
| 2004 |  |  | 30 |  |
| 2005 |  | CPI 0A | FE |  |
| 2006 |  |  | 0A |  |
| 2007 |  | JC 200D | DA |  |
| 2008 |  |  | 0D |  |
| 2009 |  |  | 20 |  |
| 200A |  | SUI 07 | D6 |  |
| 200B |  |  | 07 |  |
| 200C |  | STA 3050 | 32 |  |
| 200D |  |  | 50 |  |
| 200E |  |  | 30 |  |
| 200F |  | HLT | 76 |  |

**Experimental Results :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Input DATA** |  | **RESULT** |  |  |
| Memory location | Data | Memory location | Data | Comments |
| 2050 | **31H** | 3050 | 0BH |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Conclusion:**

**Here we successfully convert bcd number to hexadecimal .**

**Precautions:**

1. Properly connect the 8085 microprocessor kit with power supply terminals.
2. Switch on the power supply after checking connections
3. Handle the Trainer kit carefully.