Ministry of Higher Education
Egyptian Academy for Engineering & Advanced Technology (EAEAT)
Affiliated by Ministry of Military Production

Research Article

In 8086 Microprocessor

Department	Electrical engineering
Academic Year	Second Year
Course name	Microprocessor and its Applications
Course code	ECO272

<u>Title: -</u>

A survey on 8086 Microprocessor

By:

	Name	Edu mail	ID
1	Omar Mohsen Emam	o.mohssen2017015@eaeat-academy.edu.eg	2017015

Approved by:

Examiners committee	Signature

Research objectives

The objective of this Research is to make a full survey on 8086 Microprocessor, Discussing briefly its Architecture, Design pattern, Hardware Functional units and its instruction set.

Then use what we learn from its instruction we will be able to demonstrate how to make a basic calculator using assembly language and Emu8086 simulator.

Abstract

In this research we will first go through 8086 microprocessor Architecture, Features, Modes of operation, Do a small comparison between 8086 and the previous model which is 8085.

Then we will go through its functional unit from ALU, EU, BIU, Different flags, Registers and the purpose of each one of them.

Then we will discuss its instruction set and purpose of each instruction and interrupt we will use later.

Finally we will use the instructions we talked about to build basic calculator.

Program code will be attached Here.►► ► 📎

Table of contents

List of Figures

Introduction

8086 microprocessor was an advanced later version of 8085 microprocessor first introduced by Intel in 1976. It was well known for being one of the first 16-bit microprocessor chips [2] and for having 16 data lines that gives up to 1MB storage, 20 Address lines. It is also known for its strong instruction set which enhanced some core operations like multiplication and division.

The 8086 supported two modes of operation, Maximum mode to handle multiple processors and minimum mode suited for single processors.

Features of 8086 chip:

It was the first 16-bit microprocessor having a 16-bit ALU, Registers, 16-bit internal & external data bus which lead to improvement in performance speed.

It had two pipelining stage, Fetch stage and Execute stage.

Fetch stage can handle up to 6 bytes of instructions and stores them in first in first out queue to be executed in Execute stage.

It has 256 vectored interrupts.

It consists of 29,000 transistor.

Difference between 8086 and 8085:

8086 come with twice the size of 8085 chip as 8086 was 16-bit chip while 8085 was 8-bit only.

8086 had 20-bit address bus while 8085 had 16-bit bus.

8085 could access up to 64 kb memory while 8086 had up to 1Mb of memory.

8086 was distinguished by its instruction queue while 8085 doesn’t have one.

8086 supported pipelining architecture.

8085 could address 256 I/O’s, But the 8086 could access 65,536 I/O’s.

Review

Architecture of 8086:

8086 consists of two blocks BIU and EU.

BIU contains queues of

Instructions, Registers segments,

Instruction pointer, Address adder and

Address summing block [3].

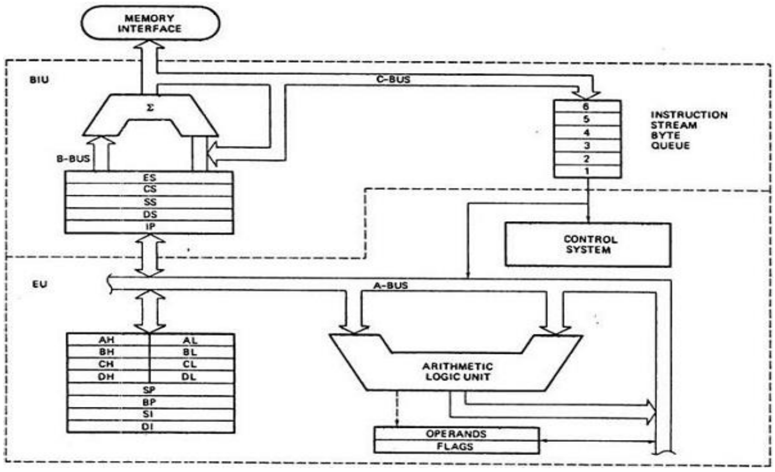


Fig (1). Architecture of 8086

EU consists of control circuitry, Instruction decoder, Pointers and Index registers, ALU, Flag register.

BIU has bus control logic which generates all bus control signals for I/O and memory such as read and write signals.

BIU handles all data transfer and address busses for execution unit.

Execution unit (EU) perform, Operates and execute instructions.

8086 Functional Units:

As expressed above 8086 is divided into two functional units EU and BIU.

EU functions is to inform BIU where to fetch instructions or data from, Decode and execute instructions.

There are other functional units connected to EU as ALU, Different flags and registers.

ALU handles all arithmetic and logical related operations.

Flag Register 16-bit unit that behave as a flip flop by changing its status according to the result stored in accumulator.

Conditional Flags represents arithmetic or logical instruction executed result, List of those flags:

- Carry flag Show an overflow condition for arithmetic operations.
- Auxiliary flag is set after an operation is resulted in carry/borrow, Also the processor uses this flag to perform binary to BCD conversion.
- Parity flag used to indicate the parity of the result, whether it's even or odd
- Zero flag set to 1 when the result of arithmetic or logical operation is zero else it is set to 0.
- Sign flag holds the sign of the result.
- Overflow flag states that system capacity is exceeded.

Control flags Controls the execution unit operation, List of those flags:

- Trap flag used for debugging as it allows single step control and allows the user to execute one instruction at a time.
- Interrupt flag used to allow/prohibit the interruption of a program.
- Direction flag used in string operation the string bytes are accessed when it is set from the higher memory address to the lower memory address and vice versa

General Purpose register are 4 16-bit registers used individually to store data each register can be divided into pairs and store 8-bit data. Those registers are AX (accumulator Register), BX (Base Register), CX (Counter Register) and DX (Data Register) [1].

Stack Pointer Register is a 16-bit register used to hold the address from the start of the segment to the location of the memory, where a word was stored recently on stack.

BIU functions is to Fetch instructions or data from memory, Write the data to memory or to port, Read data from port. It has the following functional parts

- Instruction queue as BIU got up to 6 byte of next instructions and stores them in a queue which is FIFO (first in first out) queue to be executed by EU.
- Pipelining is used to fetch next instructions while the current is being executed.
- Segment registers which holds the addresses of the instructions and data in memory, Consists of 4 segment buses CS (Code segment), DS (Data segment), SS (Stack segment) and ES (Extra segment).
- Instruction pointer 16-bit register hold address of next instruction.

Design & Simulation

Before designing our basic calculator using Emu8086 Simulator we first need to go through needed instructions from 8086 instruction set.

Instruction set:

The 8086 microprocessor supports 8 types of instructions

- Data Transfer Instructions used in transferring data from source operant to destination operant.
- Arithmetic Instructions used in performing arithmetic operations.
- Bit Manipulation Instructions in performing operations involving data bits.
- String Instructions – String is a group of bytes/words and their memory is allocated in a sequential order.
- Program Execution Transfer Instructions (Branch & Loop Instructions) are used in transferring/branching the instructions during an execution.
- Processor Control Instructions are used in controlling the processor action by setting/resetting the flag values.
- Iteration Control Instructions are used in executing the given instructions for number of times.
- Interrupt Instructions are used in calling the interrupts during program execution.

We will only go through instructions we need to do calculator

Data transfer instructions

- MOV – copy the data from source to destination provided.
- PUSH – put a word at the top of the stack.
- POP – get the word at the top of the stack.

Arithmetic Instructions

- ADD − Used in adding the provided byte to byte/word to word.
- SUB − Used in subtracting the byte from byte/word from word.
- MUL − Used in multiplying unsigned byte by byte/word by word.
- DIV − Used to divide the unsigned word by byte or unsigned double word by word.

Program Execution Transfer Instructions

- CALL − Used in calling a procedure then save their return address to the stack.
- RET − Used in returning from the procedure to the main program.
- JMP − Used in jumping to the provided address to proceed next instruction.
- JE/JZ − Used to jump if equal/zero flag ZF = 1
- JNE/JNZ − Used to jump if not equal/zero flag ZF = 0

Interrupt Instructions

- INT − Used to interrupt the program during execution and calling service specified.

Design:

Lastly we are going to use the instruction set above to create basic calculator.

Main idea is to set each arithmetic operation (Add, Subtract, Multiply, Division) to modes, Functions from which user can choose at the beginning.

To set this up we firstly need to make our main function in our case we going to name it start, then by using JMP instruction at the beginning forces program to go straight to start procedure.

After program starts, User will get the instruction message, in this message each mode of operation has a letter key to access the mode, for each operation then user will be asked to input mode preferred, After getting input from user the program then compare the input to each mode key, if input matched any mode key program goes straight to that specific operation function, if input doesn’t match any key “Choice error” message pop up.

After choosing a mode user is asked to input first number then second number after that the program takes the two inputs and do the chosen operation then print the result.

Results and Discussion

Results:

The calculator work properly, with an option to do multiple calculations without needing to restart the program.

It also takes input from user till pressing enter.

This calculator do calculations up to 65,535.

While in division mode reminder of operation will not be shown in the result.

Discussion:

This calculator do up to 65,535 since max value of 16 bit chip is 65,536.

In division mode reminder doesn’t show up because this calculator is used to count integers only so any float number will be converted to integer so reminder will be ignored.

Counter register CX is set to 10,000 during calculating the result to help reach as big output as possible but will lead to appearance of multiple zeros in the beginning of the resulting number in case of small calculations.

One of the problems encountered during simulation was the ability of user to enter a letter in the input number location and will lead to bad calculations results.

Conclusions

As a summary we was able to make a survey on 8086 microprocessor chip going through its Architecture, Features, Modes of operation, Knowing its main two blocks and what do they contain and the function of each unit in those two blocks, We compare between 8086 and 8085 and saw the difference between them, Through all that We were able to make a basic calculator throw what we learned from studying 8086 microprocessor chip and from using its instruction set through assembly language and EMU8086 simulator.

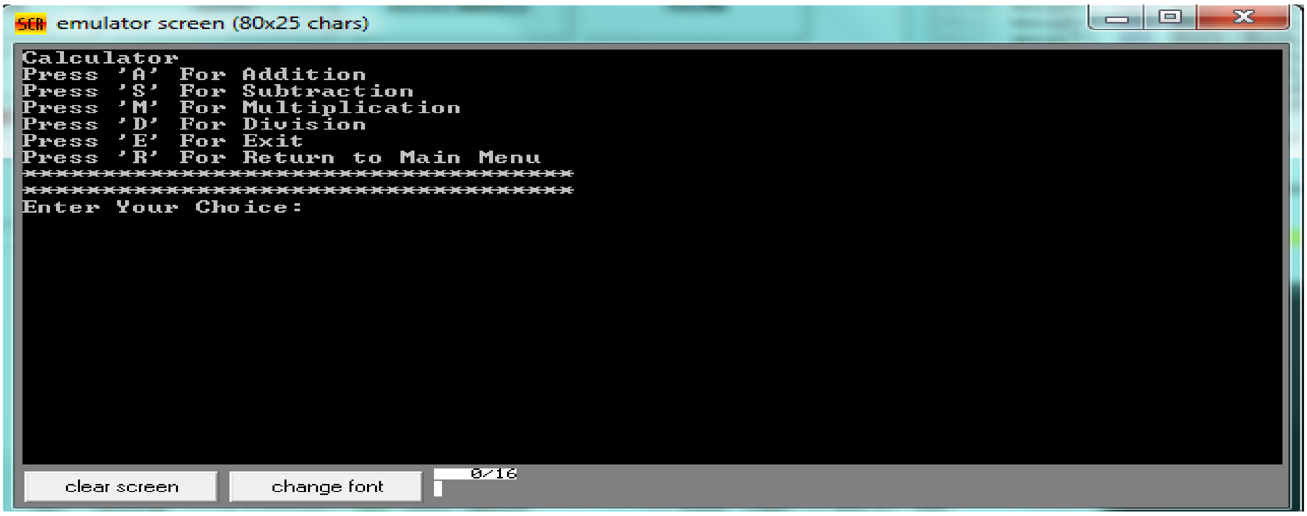
References

1 - Barry B. Berry, P54, Vernon Anthony, THE INTEL MICROPROCESSORS 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium, Pentium Pro Processor, Pentium II, Pentium III, Pentium 4, and Core2 with 64-Bit Extensions Architecture, Programming, and Interfacing, Hamilton Printing,8th edition.

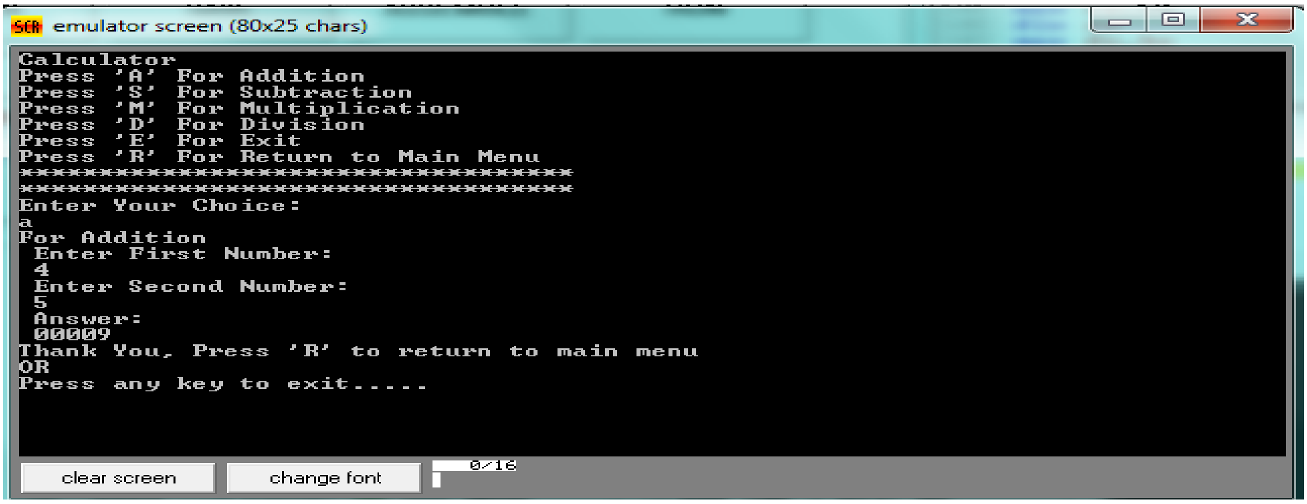
2 – PEARSON, *informIT*, [Online], By Scott Mueller and Mark Edward Soper, Published in Jun 8, 2001, viewed in May 17, 2020, URL = https://www.informit.com/articles/article.aspx?p=130978&seqNum=25

3 – TUTORIAL POINT, [online], viewed in May 17, 2020, URL = https://www.tutorialspoint.com/microprocessor/index.htm

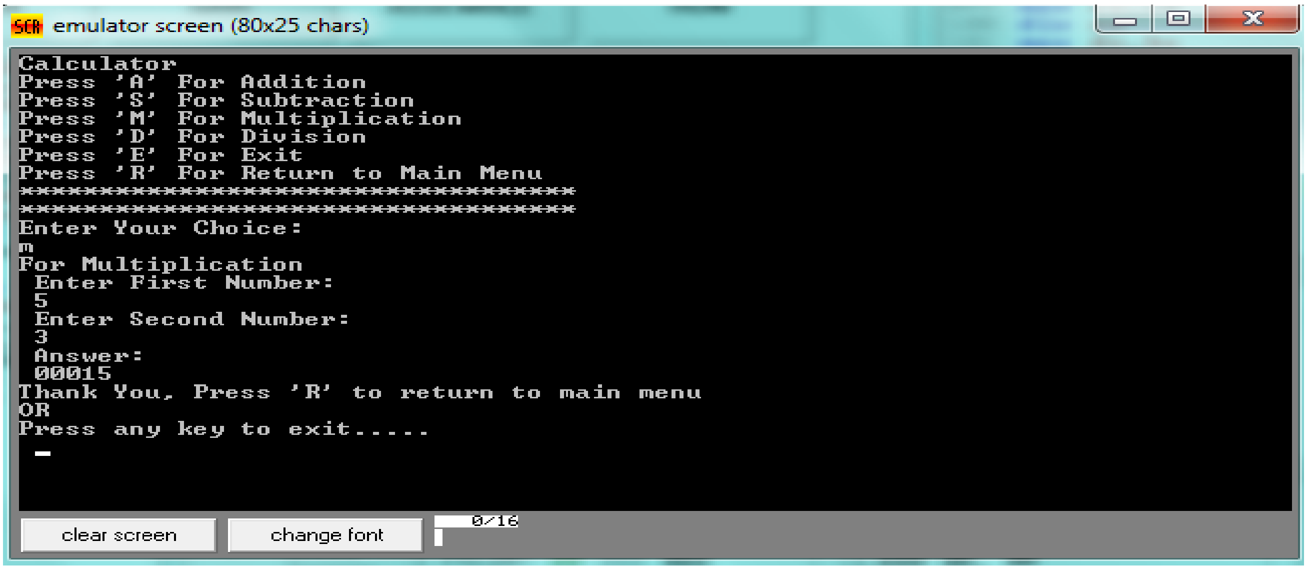
Simulation result samples:



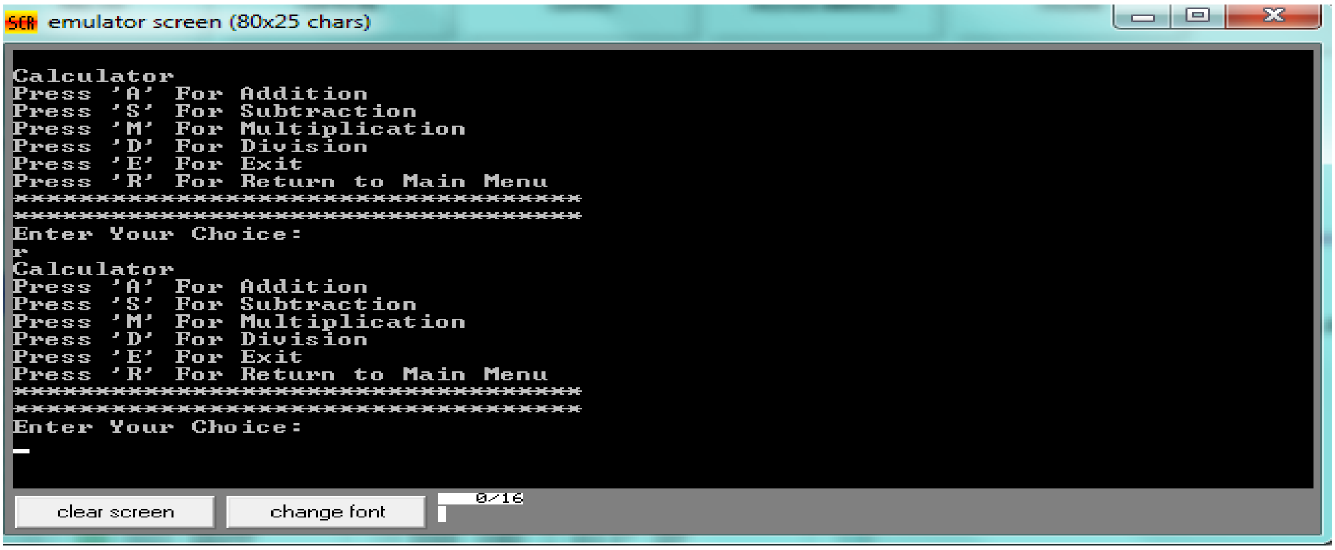
```
Calculator
Press 'A' For Addition
Press 'S' For Subtraction
Press 'M' For Multiplication
Press 'D' For Division
Press 'E' For Exit
Press 'R' For Return to Main Menu
***********************************
***********************************
Enter Your Choice:
```



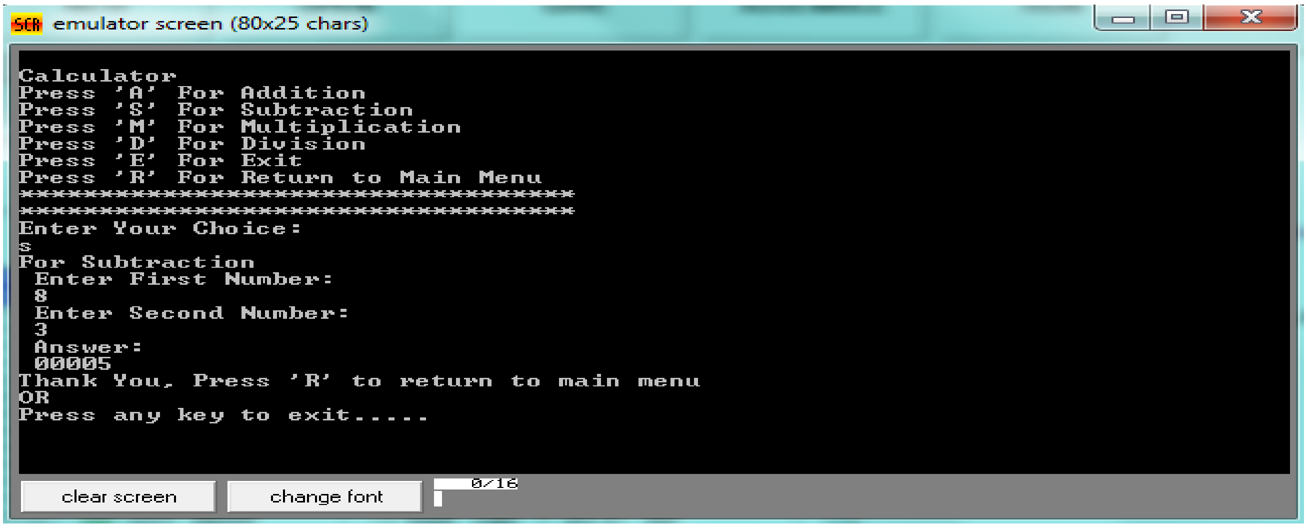
```
Calculator
Press 'A' For Addition
Press 'S' For Subtraction
Press 'M' For Multiplication
Press 'D' For Division
Press 'E' For Exit
Press 'R' For Return to Main Menu
***********************************
***********************************
Enter Your Choice:
a
For Addition
 Enter First Number:
 4
 Enter Second Number:
 5
 Answer:
 00009
Thank You, Press 'R' to return to main menu
OR
Press any key to exit.....
```



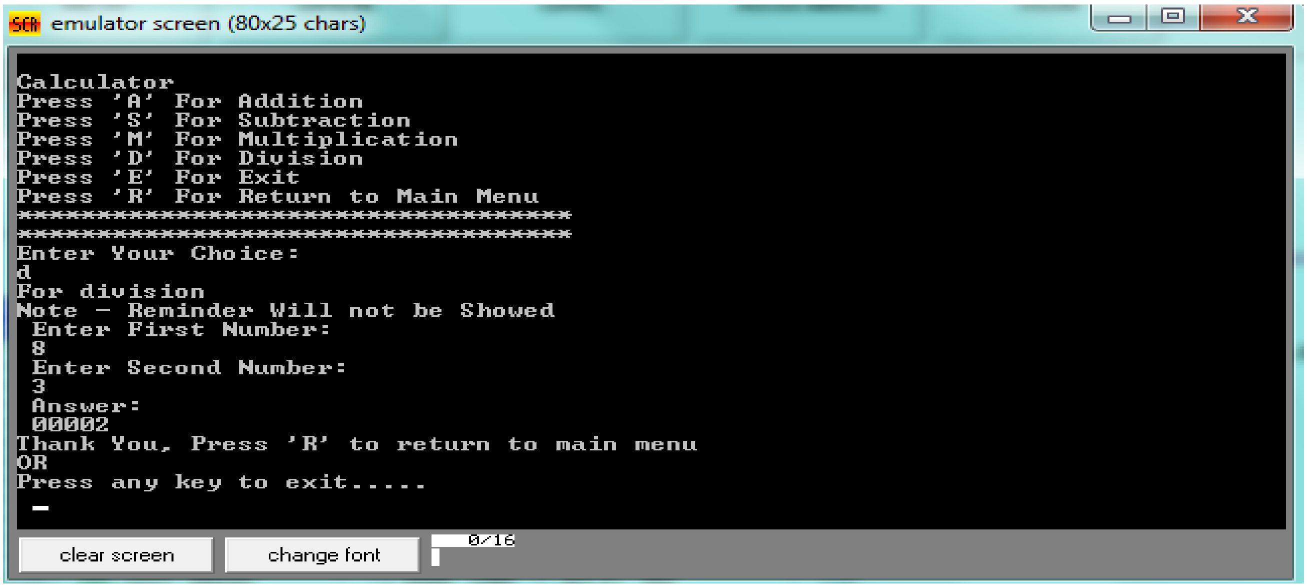
```
Calculator
Press 'A' For Addition
Press 'S' For Subtraction
Press 'M' For Multiplication
Press 'D' For Division
Press 'E' For Exit
Press 'R' For Return to Main Menu
***********************************
***********************************
Enter Your Choice:
m
For Multiplication
 Enter First Number:
 5
 Enter Second Number:
 3
 Answer:
 00015
Thank You, Press 'R' to return to main menu
OR
Press any key to exit.....
 _
```



```
Calculator
Press 'A' For Addition
Press 'S' For Subtraction
Press 'M' For Multiplication
Press 'D' For Division
Press 'E' For Exit
Press 'R' For Return to Main Menu
***********************************
***********************************
Enter Your Choice:
r
Calculator
Press 'A' For Addition
Press 'S' For Subtraction
Press 'M' For Multiplication
Press 'D' For Division
Press 'E' For Exit
Press 'R' For Return to Main Menu
***********************************
***********************************
Enter Your Choice:
_
```



```
Calculator
Press 'A' For Addition
Press 'S' For Subtraction
Press 'M' For Multiplication
Press 'D' For Division
Press 'E' For Exit
Press 'R' For Return to Main Menu
***********************************
***********************************
Enter Your Choice:
s
For Subtraction
 Enter First Number:
 8
 Enter Second Number:
 3
 Answer:
 00005
Thank You, Press 'R' to return to main menu
OR
Press any key to exit.....
```



```
Calculator
Press 'A' For Addition
Press 'S' For Subtraction
Press 'M' For Multiplication
Press 'D' For Division
Press 'E' For Exit
Press 'R' For Return to Main Menu
***********************************
***********************************
Enter Your Choice:
d
For division
Note - Reminder Will not be Showed
 Enter First Number:
 8
 Enter Second Number:
 3
 Answer:
 00002
Thank You, Press 'R' to return to main menu
OR
Press any key to exit.....
 _
```