Laboratory Manual

For

COMPUTER ORGANISATION

(IT-402)

B.Tech (IT)

SEM IV

June 2010

Faculty of Technology
Dharmsinh Desai University
Nadiad.
www.ddu.ac.in

Table of Contents

Sample Experiment

1 AIM: Program for double the number until it becomes negative using RTM.

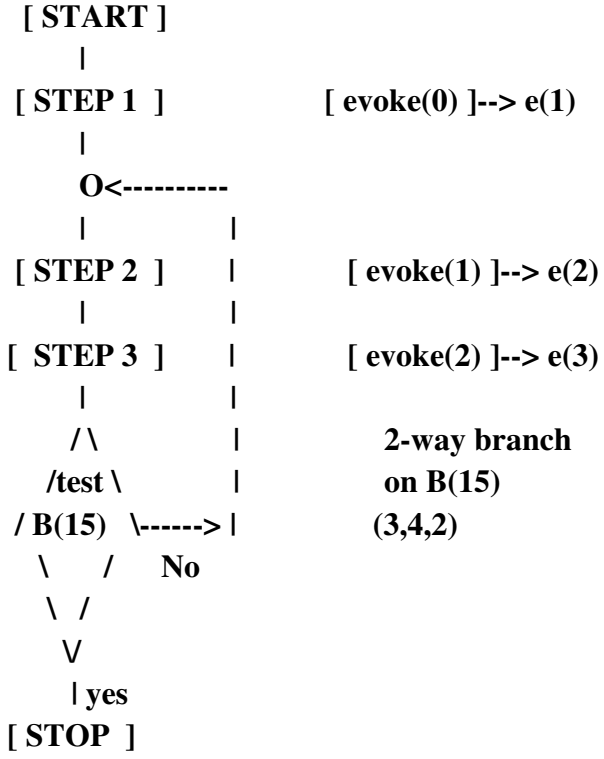
2 TOOLS: Turbo Pascal, RTM Simulator.

3 STANDARD PROCEDURES:

3.1 Analyzing the Problem:

To double number until it become negative, so we need to double until first bit of Of the number become 1.

3.2 Designing the Solution:

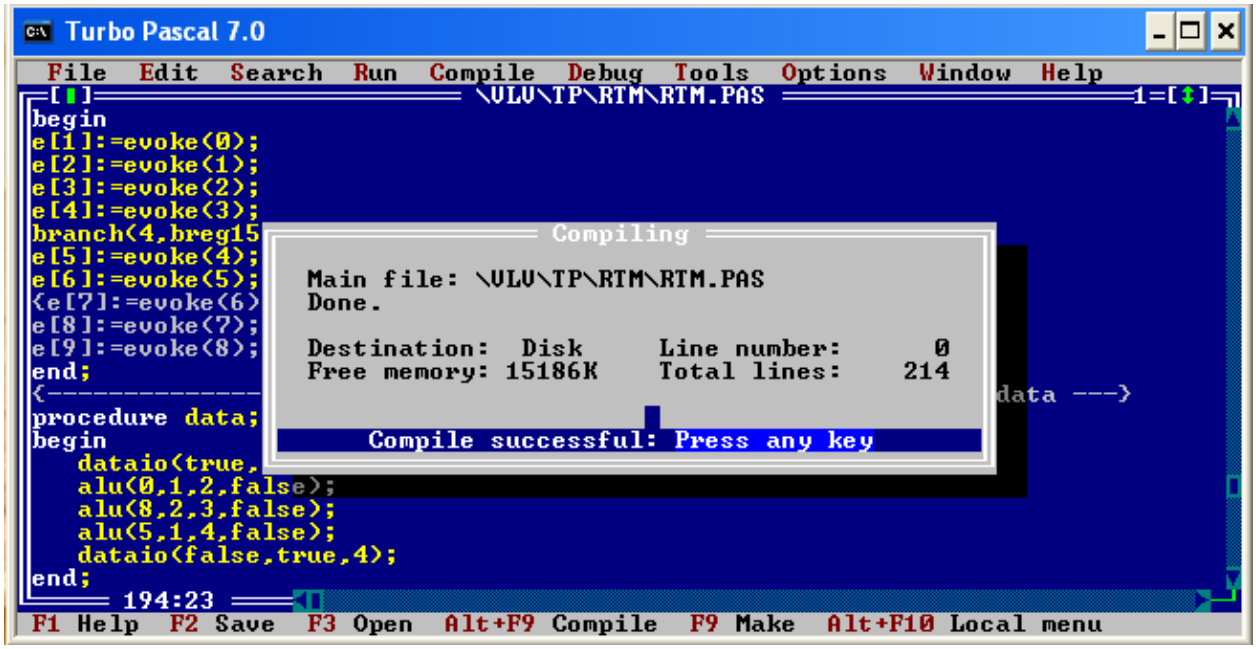


3.3 Implementing the Solution

```
{-------------------------------------------------------------------}

{====                DOUBLE.PAS                  =====}

{-------------------------------------------------------------------}

procedure control;

begin

e[1]:=evoke(0);
e[2]:=evoke(1);
e[3]:=evoke(2);
e[4]:=evoke(3);
branch(4,breg15,5,3);
e[5]:=evoke(4);
end;

{------------------------------------------------------------ data ---}

procedure data;

begin

  dataio(true,false,2);
  alu(0,1,2,false);
  alu(8,2,3,false);
  alu(5,1,4,false);
  dataio(false,true,4);
end;

{=========================================================}

{            END OF DOUBLE.PAS                          }
```

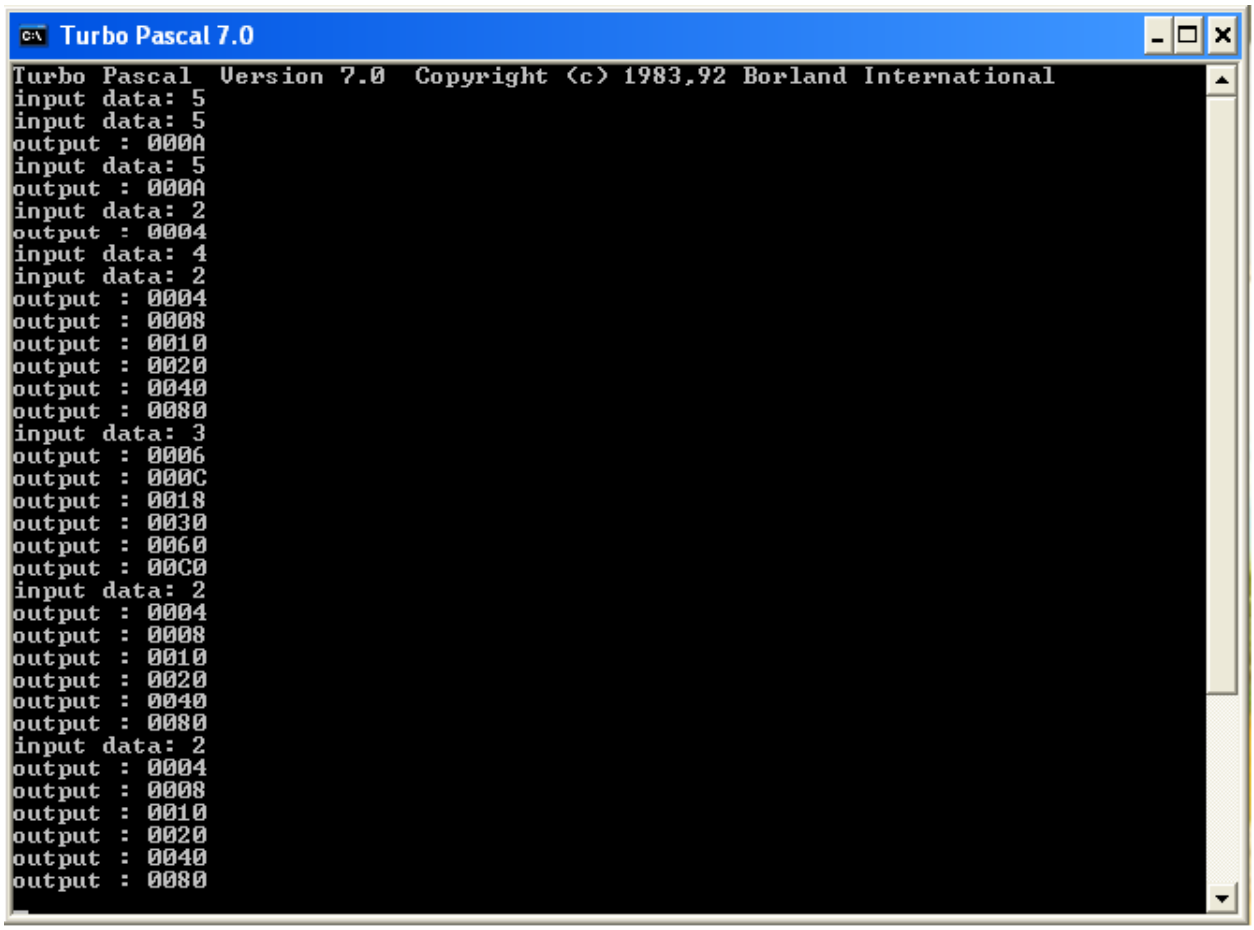
3.3.2 Compilation /Running and Debugging the Solution

For compiling the program press F9



3.4 Testing the Solution

For Running the program press ctrl+F9



4 Conclusion

So, we can conclude that input number is became negative when it's first bit become 1.

EXPERIMENT-1

Aim: Study of RTM simulator programme.
Apparatus (Software): Turbo Pascal, RTM.txt and RTM.pas files.
Procedure: Study of RTM simulator functions and instructions.
1) RTM module has two parts data part and control part
2) Module available for the control part are: evoke, branch, two way branch, eight way branch, two way merge.
3) Modules available for the data part are: dataio, ALU, constant, scratch, memory.

EXPERIMENT-2

Aim: Program for addition of two numbers using RTM.
Apparatus (Software): Turbo Pascal, RTM.txt and RTM.pas files.
Procedure: To do this program follow this steps:
1) Write two Pascal procedure data and control.
2) Then form a source file having these two procedure with a name user.pas.
3) Start Turbo Pascal and compile and execute the source file RTM.pas.
4) If there are any changes in your design, they can be incorporated by changing your data and control procedures in the source file User.pas.
5) In this addition program, we have to add two numbers taken from keyboard, store it into given registers and display it.

EXPERIMENT-3

Aim: Program for subtraction of two numbers using RTM..
Apparatus (Software): Turbo Pascal, RTM.txt and RTM.pas files.
Procedure: To do this program follow this steps:
1) Write two Pascal procedure data and control.
2) Then form a source file having these two procedure with a name user.pas.
3) Start Turbo Pascal and compile and execute the source file RTM.pas.
4) If there are any changes in your design, they can be incorporated by changing your data and control procedures in the source file User.pas.
5) In this subtraction program, we have to subtract two numbers taken from keyboard, store it into given registers and display it

EXPERIMENT-4

Aim: Program for double the number until it becomes negative using RTM.
Apparatus (Software): Turbo Pascal, RTM.txt and RTM.pas files.
Procedure: To do this program follow this steps:
1) Write two Pascal procedure data and control.
2) Then form a source file having these two procedure with a name user.pas.
3) Start Turbo Pascal and compile and execute the source file RTM.pas.
4) If there are any changes in your design, they can be incorporated by changing your data and control procedures in the source file User.pas.
5) In this addition program, we have to input any number from keyboard, double it check until it becomes negative every time display the number.

EXPERIMENT-5

Aim: Program for addition of ten numbers using RTM.
Apparatus (Software): Turbo Pascal, RTM.txt and RTM.pas files.
Procedure: To do this program follow this steps:
1) Write two Pascal procedure data and control.
2) Then form a source file having these two procedure with a name user.pas.
3) Start Turbo Pascal and compile and execute the source file RTM.pas.
4) If there are any changes in your design, they can be incorporated by changing your data and control procedures in the source file User.pas.
5) In this addition program, we have to ten numbers taken from keyboard, add them also use branch operation display the result.

EXPERIMENT-6

Aim: Program for addition of ten numbers using RTM.
Apparatus (Software): Turbo Pascal, RTM.txt and RTM.pas files.
Procedure: To do this program follow this steps:
1) Write two Pascal procedure data and control.
2) Then form a source file having these two procedure with a name user.pas.
3) Start Turbo Pascal and compile and execute the source file RTM.pas.
4) If there are any changes in your design, they can be incorporated by changing your data and control procedures in the source file User.pas.
5) In this addition program, we have to generate fibonacci series and display it.

EXPERIMENT-7

Aim: Program for implementing Booth's algorithm using 'c' language.
Apparatus (Software): Turbo c.
Procedure: To do this program follow this steps:
1) The algorithm works for positive multipliers.
2) Booth`s algorithm examination of multiplier bits and shifting of the partial product.
3) Prior to the shifting, the multiplicand may be added to the partial product, subtracted or left unchanged.

EXPERIMENT-8

Aim: Implement Up-down counter using 74193 IC.
Apparatus (Hardware): Power supply, digital trainer kit, 74193 IC,clock pulse generator, connecting wires multimeter etc
Procedure: 74193 IC is a 4-bit binary synchronous up/down counter and it has parallel load capability also
To do this experiment follow this steps:
1) According to pin assignment connections are done to 74193 IC.
2) Separate terminal count up and terminal counts down outputs are used as clock.
3) When parallel data in(PL`) and master reset(MR) inputs are low ,information present on parallel data in(P0-P3)is loaded into counter and appears on outputs.
4) This counter can operate either in count up or count down mode depending upon clock applied.
5) Waveform can be observed on CRO for count up and count down mode.

EXPERIMENT-9

Aim: Implement Johnson counter using 7474 ICs.
Apparatus (Hardware): Power supply, digital trainer kit, 7474 IC, clock pulse generator, connecting wires multimeter etc
Procedure: 7474 IC is dual D-type positive edge triggered flip-flop with direct set and direct clear inputs and complementary outputs.

1) In the Johnson counter, complementary output of the last flip-flop is connected to the input of first flip-flop.
2) As we have to implement 4-bit Johnson counter, we require two 7474 D-flip-flop ICs.
3) According to Pin assignment connections are done to 7474 ICs.
4) Waveform can be observed on CRO for outputs of four flip-flops .

EXPERIMENT-10

Aim: Implement ring counter using 7474 IC.
Apparatus (Hardware): Power supply, digital trainer kit, 7474 IC, clock pulse generator, connecting wires multimeter etc
Procedure: 7474 IC is dual D-type positive edge triggered flip-flop with direct set and direct clear inputs and complementary outputs.
1) In the ring counter, normal output of the last flip-flop is connected to the input of first flip-flop.
2) As we have to implement 4-bit ring counter, we require two 7474 D-flip-flop ICs.
3) According to Pin assignment connections are done to 7474 ICs.
4) Waveform can be observed on CRO for outputs of four flip-flops.

EXPERIMENT-11

Aim: Implement read write operation using RAM 6264 IC.

Apparatus (Hardware): RAM 6264 IC, power supply, bread board, connecting wires.

Procedure: RAM 6264 IC is 8k*8 configuration .some operation like emory read and memory write can be done using this IC.It has bi-directional data bus and unidirectional address bus.

To do this program follow this steps:

1) To perform read write operations connect the IC in following manner:

a) Memory write(read)operation takes place if WE`=0(write)WE`=1(read).

b) Chip select (CS`=0 or ground)triggers memory operation.

c) Chip enable(CE=1 or VCC)enables the chip.

d) Outputs enables(OE`)controls bi-directional data bus i.e. OE`=1(write)and OE`=0(read).

EXPERIMENT-12

Aim: Implement shift register using 74166 and 7493 ICs.
Apparatus (Hardware): Power supply, digital trainer kit, 74166 and 7493 ICs, clock pulse generator, connecting wires multimeter etc
Procedure: 74166 IC is for parallel in serial out register and 7493 IC is a 4-bit binary synchronous up-down counter.

1) In the shift register, for serial- in-serial out: output Q_B of the 7493 IC is connected to serial input of 74166 and shift/load=1
2) In the shift register, for serial- in-parallel out: output Q_D of the 7493 IC is connected to shift/load input of 74166 and serial input=0.
3) According to Pin assignment connections are done to 7493 and 74166 IC.
4) Waveform can be observed on CRO for parallel and serial outputs.

EXPERIMENT-13

Aim: Program for implementing Booth's algorithm for multiplying two negative number using 'c' Language.

Apparatus (Software): Turbo c.

Procedure: To do this program follow this steps:

1) The algorithm works for negative multipliers in 2`s complement representation.
2) Booth`s algorithm examination of multiplier bits and shifting of the partial product.
3) Prior to the shifting, the multiplicand may be added to the partial product, subtracted or left unchanged.

EXPERIMENT-14

Aim: Program for implementing gcd of two numbers using ‘c’ language.
Apparatus (Software): Turbo c.
Procedure: To do this program follow this steps:
1)Take two numbers as input.
2) Implement gcd function.
3)Output will show the gcd of two input numbers.

References

- RTM.txt is available with RTM package. It contains details about RTM
- COMPUTER ARCHITECTURE AND ORGANISATION" 3rd editionJOHN .P. HAYES, Computer science series , McGRAW-HILL
- COMPUTER SYSTEM ARCHITECTURE ", MORRIS MANO, PHI.
- COMPUTER ORGANISATION " 3rd edition, HMACHER,VRANESIC and ZAKY., Computer Science Ser ies, McGRAW- HILL