INDEX

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| SNO | DATE | LAB NO | LAB OBJECTIVE | SIGN |
| 1 | 3-11-21 | 1 | INTRODUCTION TO VVM |  |
| 2 | 3-11-21 | 2 | INTRODUCTION TO VVM PROGRAMMING |  |
| 3 | 10-11-21 | 3 | CONDITIONAL STATEMENTS |  |
| 4 | 17-11-21 | 4 | INTRODUCTION TO MIPS |  |
| 5 | 17-11-21 | 5 | MIPS ASSEMBLY LANGUAGE ARITHMETIC OPERATIONS |  |
| 6 | 24-11-21 | 6 | BITWISE OPERATION IN MIPS |  |
| 7 | 07-12-21 | 7 | IF Else & Switch Case |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

Bahria University,

Karachi Campus

A picture containing text, room

Description automatically generated

LAB EXPERIMENT NO.

**06**

LIST OF TASKS

|  |  |
| --- | --- |
| TASK NO | OBJECTIVE |
| 1 | **Complete the table by solving the bitwise instruction of all Logical gates. Add the code and output of the logical gates to show solution of MASK BITS given in the table.** |
|  |  |
|  |  |

Submitted On:

**22-Dec-2021**

**Date: DD/MM/YY**

**Task No 1:** **Complete the table by solving the bitwise instruction of all Logical gates. Add the code and output of the logical gates to show solution of MASK BITS given in the table.**

|  |  |  |
| --- | --- | --- |
| **Logic** | **Mask Bits** | |
|  | 0 | 1 |
| **AND** | 0 | 6 |
| **OR** | 6 | -1 |
| **NOT** |  |  |
| **XOR** | 6 | -7 |
| **XNOR** | -7 | 6 |
| **NOR** | -7 | 0 |
| **NAND** | -1 | -7 |

**SOLUTION**

**AND OPERTAOR:**

.data

input: .asciiz "\n ENTER AN INTEGER VALUE: " # variable declaration

result: .asciiz "\n RESULT: "

.text

li $t0,0x00000000

la $a0,input

li $v0,4

syscall

li $v0,5

syscall

move $t1,$v0

and $t2,$t1,$t0

message:

la $a0,result

li $v0,4

syscall

move $a0,$t2

li $v0,1

syscall

li $v0,10

syscall

**Output:**

Graphical user interface, text, application, email

Description automatically generated

**MASK 1:**

.data

input: .asciiz "\n ENTER AN INTEGER VALUE: "

result: .asciiz "\n RESULT: "

.text

li $t0,0xffffffff

la $a0,input

li $v0,4

syscall

li $v0,5

syscall

move $t1,$v0

and $t2,$t1,$t0

message:

la $a0,result

li $v0,4

syscall

move $a0,$t2

li $v0,1

syscall

li $v0,10

syscall

**Output:**

Text

Description automatically generated

**OR OPERTAOR:**

**MASK 0:**

.data

input: .ASCIIs "\n ENTER AN INTEGER VALUE: " # variable declaration

result: .asciiz "\n RESULT: "

.text

li $t0,0x00000000

la $a0,input

li $v0,4

syscall

li $v0,5

syscall

move $t1,$v0

or $t2,$t1,$t0

message:

la $a0,result

li $v0,4

syscall

move $a0,$t2

li $v0,1

syscall

li $v0,10

syscall

**Output:**

Graphical user interface, text, application, email

Description automatically generated

**MASK 1:**

.data

input: .asciiz "\n ENTER AN INTEGER VALUE: " # variable declaration

result: .asciiz "\n RESULT: "

.text

li $t0,0xffffffff

la $a0,input

li $v0,4

syscall

li $v0,5

syscall

move $t1,$v0

or $t2,$t1,$t0

message:

la $a0,result

li $v0,4

syscall

move $a0,$t2

li $v0,1

syscall

li $v0,10

syscall

**Output:**

Graphical user interface, text, application, email

Description automatically generated

**NOT OPERTAOR:**

**MASK 0:**

.data

input: .asciiz "\n ENTER AN INTEGER VALUE: " # variable declaration

result: .asciiz "\n RESULT: "

.text

li $t0,0x00000000

la $a0,input

li $v0,4

syscall

li $v0,5

syscall

move $t1,$v0

not $t2,$t1

message:

la $a0,result

li $v0,4

syscall

move $a0,$t2

li $v0,1

syscall

li $v0,10

syscall

**Output:**

Text

Description automatically generated

**MASK 1:**

.data

input: .asciiz "\n ENTER AN INTEGER VALUE: " # variable declaration

result: .asciiz "\n RESULT: "

.text

li $t0,0xffffffff

la $a0,input

li $v0,4

syscall

li $v0,5

syscall

move $t1,$v0

not $t2,$t1

message:

la $a0,result

li $v0,4

syscall

move $a0,$t2

li $v0,1

syscall

li $v0,10

syscall

**Output:**

Text

Description automatically generated

**XOR OPERTAOR:**

**MASK 0:**

.data

input: .asciiz "\n ENTER AN INTEGER VALUE: " # variable declaration

result: .asciiz "\n RESULT: "

.text

li $t0,0x00000000

la $a0,input

li $v0,4

syscall

li $v0,5

syscall

move $t1,$v0

xor $t2,$t1,$t0

message:

la $a0,result

li $v0,4

syscall

move $a0,$t2

li $v0,1

syscall

li $v0,10

syscall

**OUTPUT:**

Text

Description automatically generated

**MASK 1:**

.data

input: .asciiz "\n ENTER AN INTEGER VALUE: " # variable declaration

result: .asciiz "\n RESULT: "

.text

li $t0,0xffffffff

la $a0,input

li $v0,4

syscall

li $v0,5

syscall

move $t1,$v0

xor $t2,$t1,$t0

message:

la $a0,result

li $v0,4

syscall

move $a0,$t2

li $v0,1

syscall

li $v0,10

syscall

**OUTPUT:**

Graphical user interface, text, application, email

Description automatically generated

**NOR OPERTAOR:**

**MASK 0:**

.data

input: .asciiz "\n ENTER AN INTEGER VALUE: " # variable declaration

result: .asciiz "\n RESULT: "

.text

li $t0,0x00000000

la $a0,input

li $v0,4

syscall

li $v0,5

syscall

move $t1,$v0

nor $t2,$t1,$t0

message:

la $a0,result

li $v0,4

syscall

move $a0,$t2

li $v0,1

syscall

li $v0,10

syscall

**OUTPUT:**

Text

Description automatically generated

**MASK 1:**

.data

input: .asciiz "\n ENTER AN INTEGER VALUE: " # variable declaration

result: .asciiz "\n RESULT: "

.text

li $t0,0xffffffff

la $a0,input

li $v0,4

syscall

li $v0,5

syscall

move $t1,$v0

nor $t2,$t1,$t0

message:

la $a0,result

li $v0,4

syscall

move $a0,$t2

li $v0,1

syscall

li $v0,10

syscall

**OUTPUT:**

Text

Description automatically generated

**NAND OPERTAOR:**

**MASK 0:**

.data

input: .asciiz "\n ENTER AN INTEGER VALUE: " # variable declaration

result: .asciiz "\n RESULT: "

.text

li $t0,0x00000000

la $a0,input

li $v0,4

syscall

li $v0,5

syscall

move $t1,$v0

and $t2,$t1,$t0

not $t3,$t2

message:

la $a0,result

li $v0,4

syscall

move $a0,$t3

li $v0,1

syscall

li $v0,10

syscall

**OUTPUT:**

Graphical user interface, text

Description automatically generated

**MASK 1:**

.data

input: .asciiz "\n ENTER AN INTEGER VALUE: " # variable declaration

result: .asciiz "\n RESULT: "

.text

li $t0,0xffffffff

la $a0,input

li $v0,4

syscall

li $v0,5

syscall

move $t1,$v0

and $t2,$t1,$t0

not $t3,$t2

message:

la $a0,result

li $v0,4

syscall

move $a0,$t3

li $v0,1

syscall

li $v0,10

syscall

**OUTPUT:**

Graphical user interface, text, application, email

Description automatically generated

**XNOR OPERTAOR:**

**MASK 0:**

.data

input: .asciiz "\n ENTER AN INTEGER VALUE: " # variable declaration

result: .asciiz "\n RESULT: "

.text

li $t0,0x00000000

la $a0,input

li $v0,4

syscall

li $v0,5

syscall

move $t1,$v0

xnor $t2,$t1,$t0

message:

la $a0,result

li $v0,4

syscall

move $a0,$t2

li $v0,1

syscall

li $v0,10

syscall

**OUTPUT:**

Text

Description automatically generated

**MASK 1:**

.data

input: .asciiz "\n ENTER AN INTEGER VALUE: " # variable declaration

result: .asciiz "\n RESULT: "

.text

li $t0,0xffffffff

la $a0,input

li $v0,4

syscall

li $v0,5

syscall

move $t1,$v0

xor $t2,$t1,$t0

not $t3,$t2

message:

la $a0,result

li $v0,4

syscall

move $a0,$t3

li $v0,1

syscall

li $v0,10

syscall

**OUTPUT:**

Graphical user interface, text

Description automatically generated