Lab 5

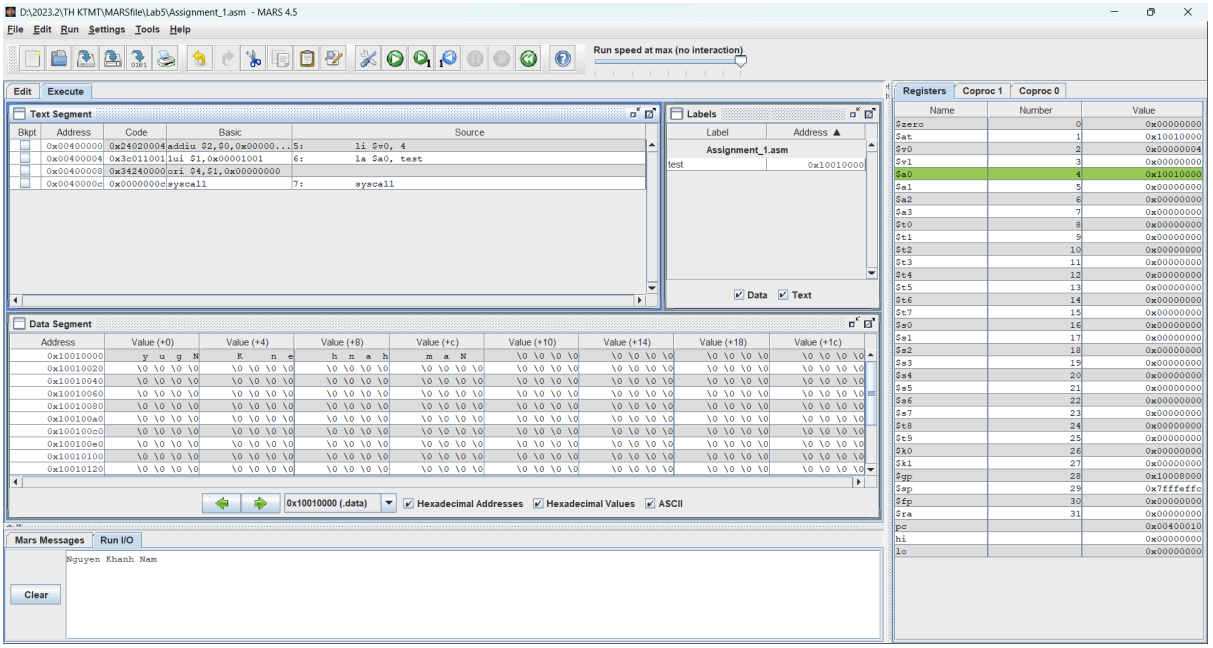
Nguyễn Khánh Nam - 20225749

Assignment 1

Code:

```
    #Laboratory Exercise 5, Assignment 1
.data
test:    .asciiz "Nguyen Khanh Nam"
.text
li       $v0,     4
la       $a0,     test
syscall
```

Result:



- Địa chỉ của string được lưu vào thanh ghi $a0

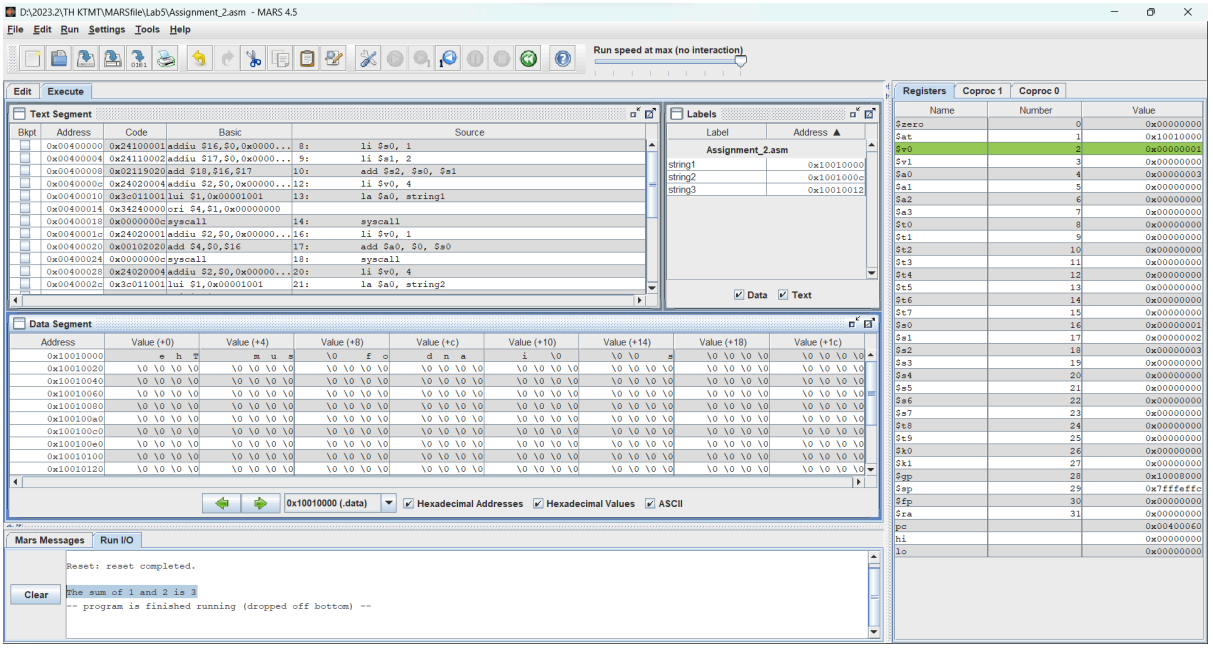
Assignment 2

Code:

```
    #Laboratory Exercise 5, Assignment 2
.data
string1:     .asciiz "The sum of "
string2:     .asciiz " and "
string3:     .asciiz " is "
```

```
.text
li      $s0,    1
li      $s1,    2
add     $s2,    $s0,        $s1

li      $v0,    4
la      $a0,    string1
syscall

li      $v0,    1
add     $a0,    $0,         $s0
syscall

li      $v0,    4
la      $a0,    string2
syscall

li      $v0,    1
add     $a0,    $0,         $s1
syscall

li      $v0,    4
la      $a0,    string3
syscall

li      $v0,    1
add     $a0,    $0,         $s2
syscall
```

Result:



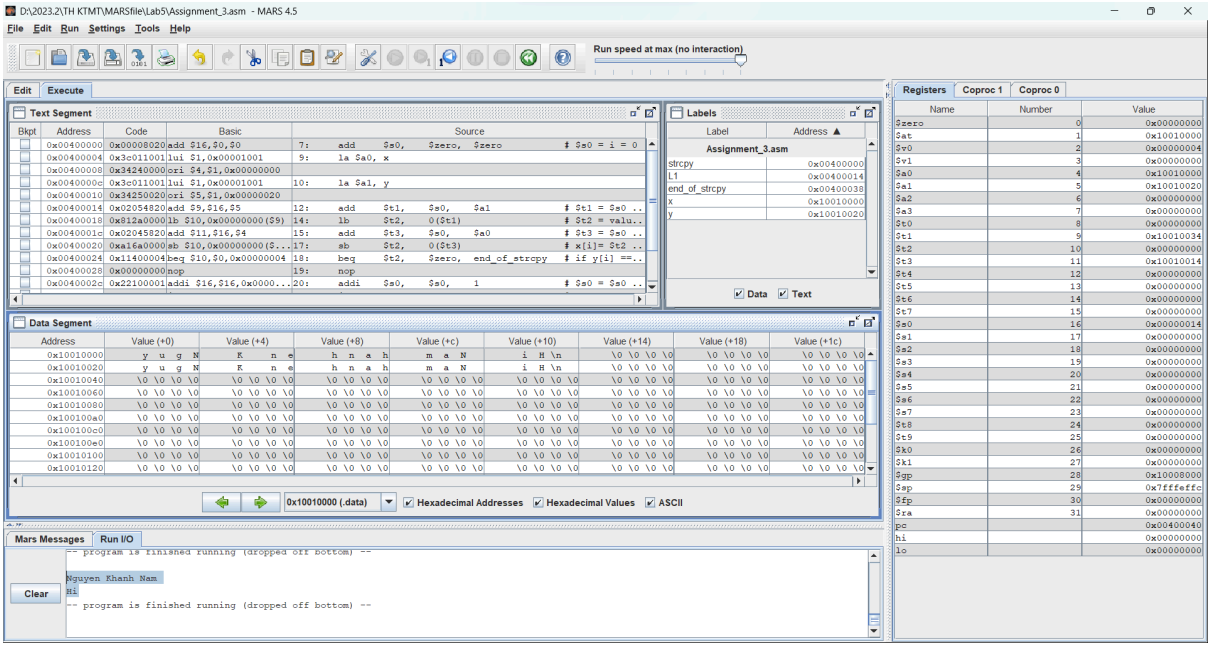
Assignment 3

Code:

```
    #Laboratory Exercise 5, Home Assignment 2
.data
x:   .space  32                          # destination string x,
empty
y:   .asciiz "Nguyen Khanh Nam \nHi"                    # source
string y
.text
strcpy:
    add      $s0,     $zero,   $zero          # $s0 = i = 0

    la $a0, x
    la $a1, y
L1:
    add      $t1,     $s0,     $a1            # $t1 = $s0 + $a1 = i +
y[0]
    # = address of y[i]
    lb       $t2,     0($t1)                  # $t2 = value at $t1 = y[i]
    add      $t3,     $s0,     $a0            # $t3 = $s0 + $a0 = i +
x[0]
    # = address of x[i]
    sb       $t2,     0($t3)                  # x[i]= $t2 = y[i]
    beq      $t2,     $zero,   end_of_strcpy  # if y[i] == 0, exit
    nop
    addi     $s0,     $s0,     1              # $s0 = $s0 + 1 <-> i = i +
1
```

```
    j       L1                                  # next character
    nop
end_of_strcpy:

    li $v0, 4
    syscall
```

Result:



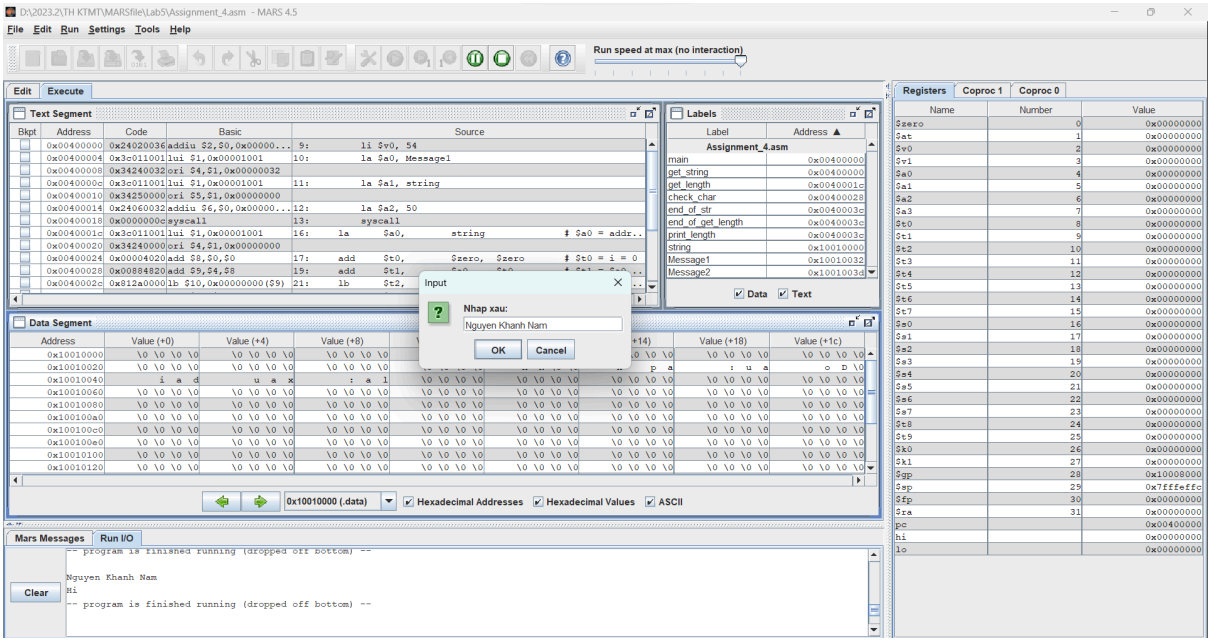
Assignment 4

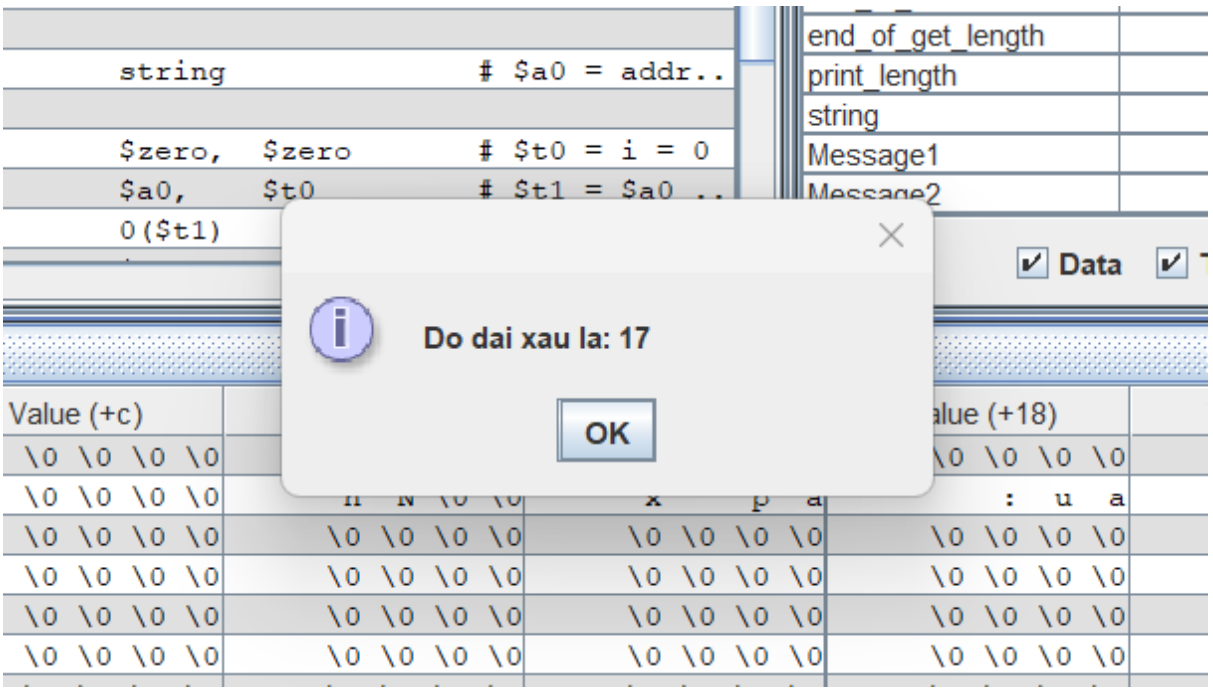
Code:

```
    #Laboratory Exercise 5, Home Assignment 3
.data
    string:     .space  50
    Message1:   .asciiz "Nhap xau: "
    Message2:   .asciiz "Do dai xau la: "
.text
main:
get_string:                                     # TODO
    li $v0, 54
    la $a0, Message1
    la $a1, string
    la $a2, 50
    syscall

get_length:
    la      $a0,        string              # $a0 = address(string[0])
    add     $t0,        $zero,  $zero       # $t0 = i = 0
```

```
check_char:
    add     $t1,        $a0,    $t0         # $t1 = $a0 + $t0
    # = address(string[i])
    lb      $t2,        0($t1)              # $t2 = string[i]
    beq     $t2,        $zero,  end_of_str  # is null char?
    addi    $t0,        $t0,    1           # $t0 = $t0 + 1 -> i = i +
1
    j       check_char
end_of_str:
end_of_get_length:
print_length:                               # TODO

    li $v0, 56
    la $a0, Message2
    add $a1, $0, $t0
    syscall
```

Result:




string # $a0 = addr..
$zero, $zero # $t0 = i = 0
$a0, $t0 # $t1 = $a0 ..
0($t1)
end_of_get_length
print_length
string
Message1
Message2
Data
Do dai xau la: 17
OK
Value (+c)
alue (+18)

Assignment 5

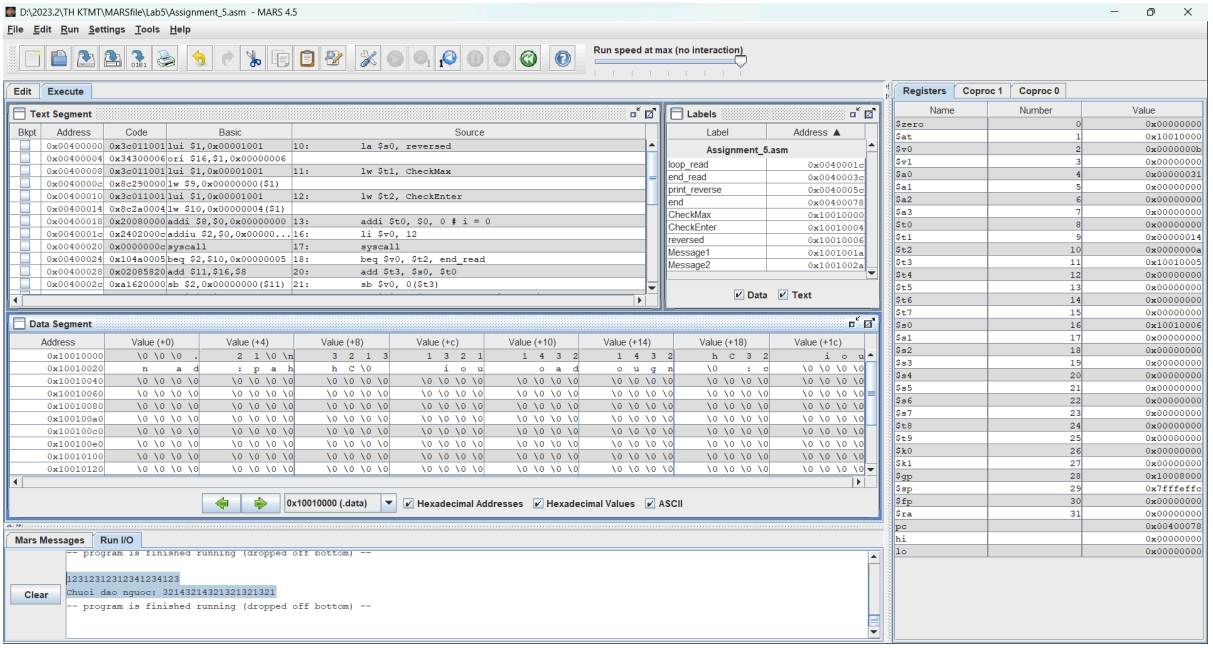
Code:

```
    #Laboratory Exercise 5, Assignment 4

.data
CheckMax:   .word   20
CheckEnter: .asciiz "\n"
reversed:   .space  20

Message2:   .asciiz "Chuoi dao nguoc: "
.text
    la      $s0,            reversed
    lw      $t1,            CheckMax
    lw      $t2,            CheckEnter
    addi    $t0,            $0,         0           # i = 0

loop_read:
    li      $v0,            12
    syscall
    beq     $v0,            $t2,        end_read
    #Load to string
    add     $t3,            $s0,        $t0
    sb      $v0,            0($t3)

    addi    $t0,            $t0,        1           #Count i

    bne     $t0,            $t1,        loop_read
    nop
end_read:
    li      $v0,            11
    lb      $a0,            CheckEnter
    syscall

    li      $v0,            4
    la      $a0,            Message2
    syscall
print_reverse:

    li      $v0,            11
    lb      $a0,            0($t3)
    syscall
```

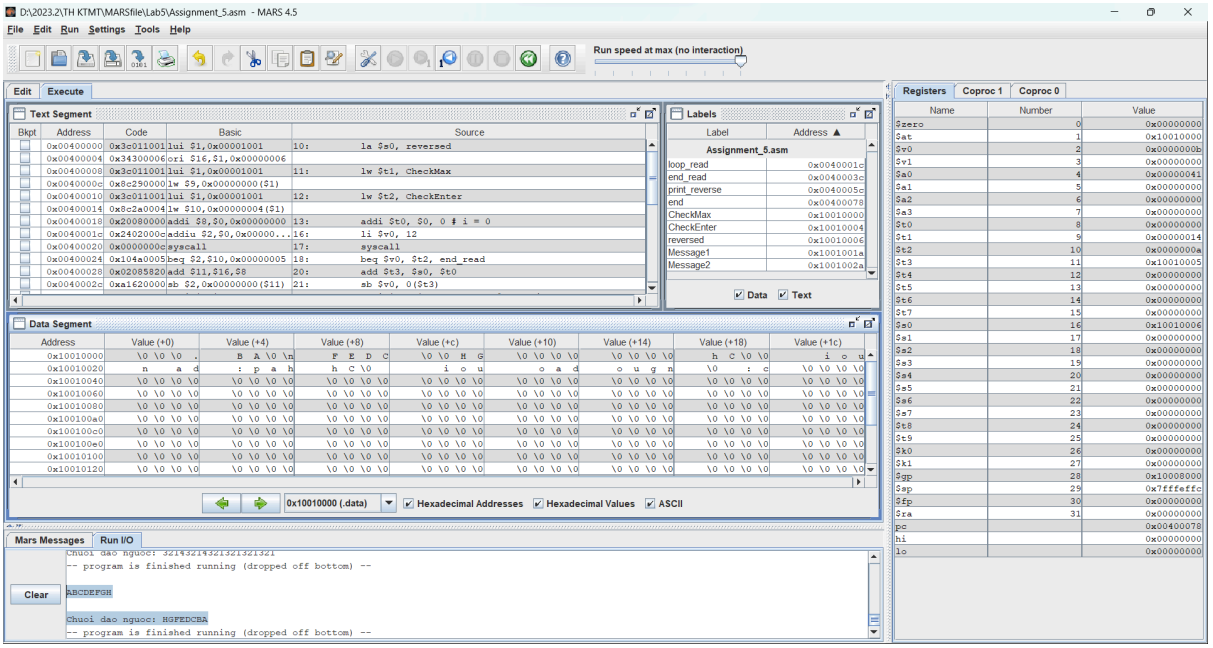
```
    add     $t3,            $t3,        -1
    addi    $t0,            $t0,        -1
    blez    $t0,            end

    j       print_reverse

end:
```

Result:

- Max 20 characters -> Stop



- Enter -> Stop


D:\2023.2\TH KTMT\MARSfile\Lab5\Assignment_5.asm - MARS 4.5
File Edit Run Settings Tools Help
Run speed at max (no interaction)
Edit Execute
Text Segment
Bkpt Address Code Basic Source
0x00400000 0x3c011001 lui $1,0x00001001 10: la $s0, reversed
0x00400004 0x34300006 ori $16,$1,0x00000006
0x00400008 0x3c011001 lui $1,0x00001001 11: lw $t1, CheckMax
0x0040000c 0x8c2500001w $9,0x00000000($1)
0x00400010 0x3c011001 lui $1,0x00001001 12: lw $t2, CheckEnter
0x00400014 0x8c2a0004 lw $10,0x00000004($1)
0x00400018 0x20080000 addi $8,$0,0x00000000 13: addi $t0, $0, 0 # i = 0
0x0040001c 0x2402000c addiu $2,$0,0x00000... 16: li $v0, 12
0x00400020 0x0000000c syscall 17: syscall
0x00400024 0x104a0005 beq $2,$10,0x00000005 18: beq $v0, $t2, end_read
0x00400028 0x02085820 add $11,$16,$8 20: add $t3, $s0, $t0
0x0040002c 0xa1620000 sb $2,0x00000000($11) 21: sb $v0, 0($t3)
Labels
Label Address
Assignment_5.asm
loop_read 0x0040001c
end_read 0x0040003c
print_reverse 0x0040005c
end 0x00400078
CheckMax 0x10010000
CheckEnter 0x10010004
reversed 0x10010006
Message1 0x1001001a
Message2 0x1001002a
Data Segment
Address Value (+0) Value (+4) Value (+8) Value (+c) Value (+10) Value (+14) Value (+18) Value (+1c)
0x10010000 \0 \0 \0 . B A \0 \n F E D C \0 \0 H G \0 \0 \0 \0 \0 \0 \0 \0 h C \0 \0 i o u
0x10010020 n a d : p a h h C \0 i o u o a d o u g n \0 : c \0 \0 \0 \0
0x10010040 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0
0x10010060 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0
0x10010080 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0
0x100100a0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0
0x100100c0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0
0x100100e0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0
0x10010100 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0
0x10010120 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0 \0
0x10010000 (.data)
Hexadecimal Addresses
Hexadecimal Values
ASCII
Mars Messages Run I/O
Chuoi dao nguoc: 3214321432132132132
-- program is finished running (dropped off bottom) --
Clear
ABCDEFGH
Chuoi dao nguoc: HGFEDCBA
-- program is finished running (dropped off bottom) --
Registers Coproc 1 Coproc 0
Name Number Value
$zero 0 0x00000000
$at 1 0x10010000
$v0 2 0x0000000b
$v1 3 0x00000000
$a0 4 0x00000041
$a1 5 0x00000000
$a2 6 0x00000000
$a3 7 0x00000000
$t0 8 0x00000000
$t1 9 0x00000014
$t2 10 0x0000000a
$t3 11 0x10010005
$t4 12 0x00000000
$t5 13 0x00000000
$t6 14 0x00000000
$t7 15 0x00000000
$s0 16 0x10010006
$s1 17 0x00000000
$s2 18 0x00000000
$s3 19 0x00000000
$s4 20 0x00000000
$s5 21 0x00000000
$s6 22 0x00000000
$s7 23 0x00000000
$t8 24 0x00000000
$t9 25 0x00000000
$k0 26 0x00000000
$k1 27 0x00000000
$gp 28 0x10008000
$sp 29 0x7fffeffc
$fp 30 0x00000000
$ra 31 0x00000000
pc 0x00400078
hi 0x00000000
lo 0x00000000