**Department of Computer Engineering**

Digital Hardware Systems

*CpeA 3104 - Microprocessors*

**Laboratory Report**

| | | | |
|---|---|---|---|
| **Laboratory Exercise No:** | **37** | **Date Performed:** | **09/23/2020** |
| **Laboratory Exercise Title:** | **Assembly Language Instructions** | | |
| **Name of Student :** | **Felisarta, German III** | **Document Version:** | **v1.0** |

## ACTIVITY #7.1

Table 1. Instructions and Outputs of Act 1.1

| Instruction | Register value after the instruction is executed |
|---|---|
| MOV AX, OFFSET STRING | **AX:** FFB8 |
| MOV DS, AX | **DS:** FFB8 |
| MOV CX, 0 | **CX:** 00 00 |
| MOV AX, [SI] | **AX**: 00 00 |
| PISH [SI] | **SP:** FFEF |
| INC SI | SI : 00 00 |
| INC CX | CX : 00 01 |

  

**ACTIVITY #7.2a**

*CODE*

```
ORG 100H

STRING1 DB "THIS IS A SAMPLE STRING $"
STRING2 DB "TH1S 1S 4 S4MPL3 STR1NG $"
STRING3 DB "NO NUMBER $"
STRING4 DB "HAS NUMBER $"

START:

    MOV DH, 0
    MOV DL, 0
    CALL SET_CURSOR_POS

    LEA DX, STRING2   ;change to either STRING1 or STRING2
    CALL DISP_MESS

    CLD
    LEA SI, STRING2   ;change to either STRING1 or STRING2
    CALL COMPARE

    JMP END_PROG


COMPARE:

    LODSB

    CMP AL, "1"
    JZ HAS
    CMP AL, "2"
    JZ HAS
    CMP AL, "3"
    JZ HAS
    CMP AL, "4"
    JZ HAS
    CMP AL, "5"
    JZ HAS
    CMP AL, "6"
    JZ HAS
    CMP AL, "7"
    JZ HAS
    CMP AL, "8"
    JZ HAS
    CMP AL, "9"
    JZ HAS
    CMP AL, "0"
    JZ HAS
```

```
    CMP AL, "$"
    JZ NONE

    JMP COMPARE

HAS:

    MOV DH, 05
    MOV DL, 0
    CALL SET_CURSOR_POS

    MOV DX, OFFSET STRING4
    CALL DISP_MESS

    RET


NONE:

    MOV DH, 05
    MOV DL, 0
    CALL SET_CURSOR_POS

    MOV DX, OFFSET STRING3
    CALL DISP_MESS

    RET


DISP_MESS:

    MOV AH, 09H
    INT 21H
    RET


SET_CURSOR_POS:

    MOV AH, 02H     ; SET CURSOR POSITION
    MOV BH, 00H
    INT 10H
    RET


END_PROG:

    END
```


Fig 1. Screenshot of HAS NUMBER


Fig 2. Screenshot of NO NUMBER

**ACTIVITY #7.2a**

*CODE*

```
ORG 100H

STRING1 DB "DAD$"
STRING2 DB "DADS$"
S_SIZE = $ - m1 db 0Dh,0Ah,'$'


START:

    MOV DH, 0
    MOV DL, 0
    CALL SET_CURSOR_POS

    CALL COMPARE

    JMP END_PROG



COMPARE:

    MOV AH, 9
    MOV DX, OFFSET STRING2
    INT 21h


    LEA DI, STRING1
    MOV SI, DI
    ADD SI, S_SIZE
    DEC SI

    MOV CX, S_SIZE
    CMP CX, 1
    JE IS

    SHR cx, 1

NEXT_CHAR:

    MOV AL, [DI]
    MOV BL, [SI]
    CMP AL, BL
    JNE NOTP
    INC DI
    DEC SI
    LOOP NEXT_CHAR
```

```
IS:

    STRING3 DB "PALINDROME $"

    MOV DH, 05
    MOV DL, 0
    CALL SET_CURSOR_POS

    MOV DX, OFFSET STRING4
    CALL DISP_MESS

    RET



NOTP:

    STRING4 DB "NOT PALINDROME $"

    MOV DH, 05
    MOV DL, 0
    CALL SET_CURSOR_POS

    MOV DX, OFFSET STRING3
    CALL DISP_MESS

    RET


DISP_MESS:

    MOV AH, 09H
    INT 21H
    RET


SET_CURSOR_POS:

    MOV AH, 02H     ; SET CURSOR POSITION
    MOV BH, 00H
    INT 10H
    RET


END_PROG:

    END
```

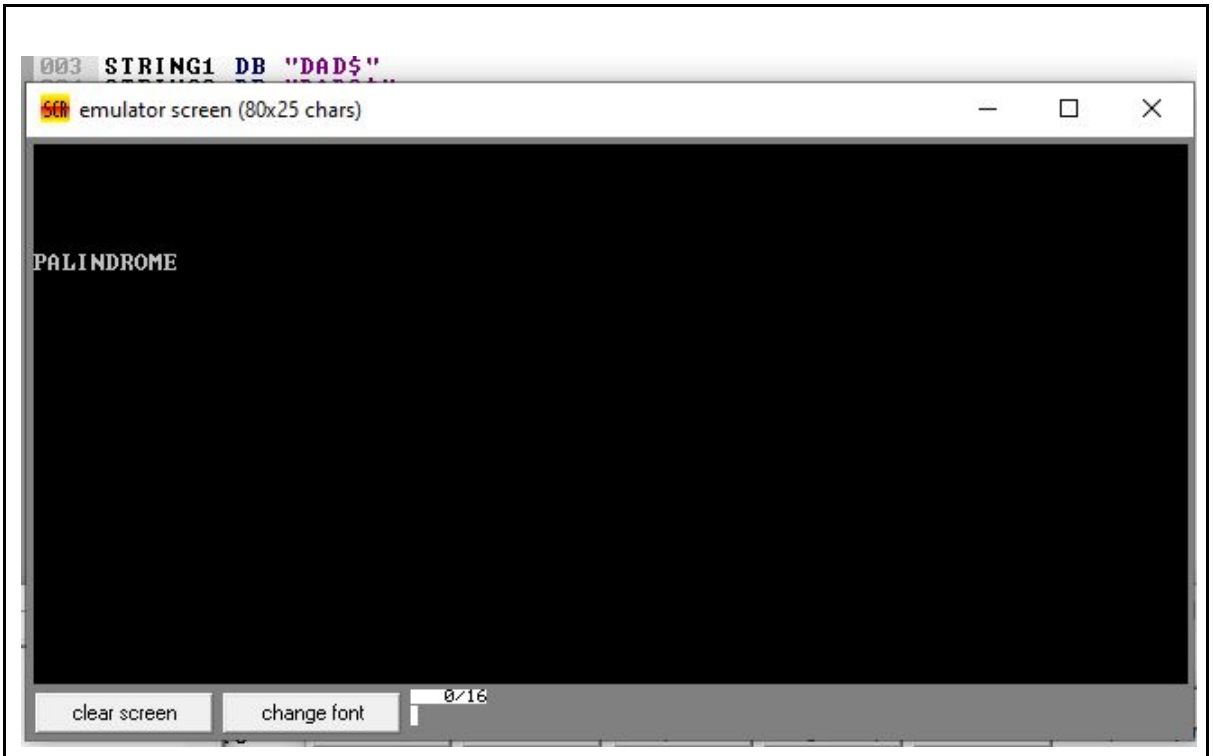


Fig 1. Screenshot of RESULT