|  |
| --- |
| PRACTICE EXERCISES OF THE MICROPROCESSORS & MICROCONTROLLERS  Instructor: The Tung Than  Student's name: Trần Lê Minh Đăng  Student code: 21520684 |

PRACTICE REPORT NO 6

PROCESSING, CALCULATION AND MEMORY ON THE 8086

MICROPROCESSOR

1. **Flowchart of the above request processing algorithm.**

**- Giải thuật tìm số Fibonacci**

B1: Tạo biến N là số Fibo cần tìm

B2: Sử dụng 3 biến S1, S2, S3

B3: Khởi tạo giá trị S1 = 0, S2 = 1, S3

B4: N <=0 thì tới B9

B5: S3 = S1

B6: S1 = S1 + S2

B7: S2 = S3

B8: N = N – 1, quay lại B4

B9: Kết thúc

A picture containing text, diagram, technical drawing, plan

Description automatically generated

1. **Explain how the algorithm works, accompanied by a video (send a Google Drive link) to demonstrate the circuit operation in case the instructor cannot run the design file.**

Link: https://drive.google.com/drive/folders/1KsxMfc27DKhFKsino96lrRaNtbjdYfkG?usp=sharing

SOURCE CODE:

|  |  |
| --- | --- |
| Code | Comment |
| .MODEL SMALL  .STACK 100H  .DATA  STR1 DB "A= $"  STR2 DB 10,13, "B= $"  STR3 DB 10,13, "FIBO: $"  X DW ?  S1 DB 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0  S2 DB 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1  S3 DB 0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0  .CODE  ;;;;;;MAIN;;;;;;;;;;;;;;;;;;  MAIN PROC  MOV AX, @DATA  MOV DS, AX  MOV AH, 9  LEA DX, STR1  INT 21H  CALL INPUT  PUSH BX  MOV AH, 9  LEA DX, STR2  INT 21H  CALL INPUT  POP CX  ADD BX,CX  MOV X,BX  MOV AH, 9  LEA DX, STR3  INT 21H  CALL PRINT\_ENTER  FIBO:  MOV AX,X  SUB AX,1  MOV X,AX  CALL S1\_TO\_S3 ;;S3 = S1  CALL ADD\_FIBO ;;S1 = S1 + S2  CALL S3\_TO\_S2 ;;S2 = S3  CALL PRINT\_FIBO ;;PRINT S1  CALL PRINT\_ENTER  MOV CX,X  ADD CX,1  LOOP FIBO  JMP END\_PROGRAM  MAIN ENDP  ;;;;;;;;NHAP SO;;;;;;;;;  INPUT PROC  MOV BX,0  LOOP1:  MOV AH,1  INT 21H  CMP AL,13  JE EXIT  SUB AL,30H  MOV AH,0  PUSH AX  MOV AX, 10  MUL BX  MOV BX, AX  POP AX  ADD BX,AX  JMP LOOP1  EXIT:  RET  INPUT ENDP  ;;;;;;;;S3 = S1;;;;;;;;;;;;  S1\_TO\_S3 PROC  MOV CX,21  S1\_1:  MOV SI,CX  MOV BL,S1[SI]  MOV S3[SI],BL  LOOP S1\_1  RET  S1\_TO\_S3 ENDP  ;;;;;;;;;S2 = S3;;;;;;;;;;;  S3\_TO\_S2 PROC  MOV CX,21  S2\_1:  MOV SI,CX  MOV BL,S3[SI]  MOV S2[SI],BL  LOOP S2\_1  RET  S3\_TO\_S2 ENDP  ;;;;;;;;S1+S2;;;;;;;;;;  ADD\_FIBO PROC  MOV CX,21  MOV AX,0  ADD\_1:  MOV BL,0  MOV SI,CX  ADD BL,AL  ADD BL,S1[SI]  ADD BL,S2[SI]  MOV AH,0  MOV AL,BL  MOV BL,10  DIV BL  MOV S1[SI],AH  LOOP ADD\_1  RET  ADD\_FIBO ENDP  ;;;;;;;;PRINT;;;;;;;;;;;  PRINT\_FIBO PROC  MOV CX,21  MOV SI,0  PRINT:  MOV DL,S1[SI]  ADD DL,48  MOV AH,2  INT 21H  INC SI  LOOP PRINT  RET  PRINT\_FIBO ENDP  ;;;;;;;;ENTER;;;;;;;;;;;  PRINT\_ENTER PROC  MOV DL,13  MOV AH,2  INT 21H  MOV DL,10  MOV AH,2  INT 21H  RET  PRINT\_ENTER ENDP  ;;;;;;;;END;;;;;;;;;;;;;  END\_PROGRAM:  END | ;;  ;;  ;;Khai báo biến  ;;  ;;  ;;  ;; X là N đề bài yêu cầu  ;;  ;;  ;;  ;;  ;;  ;;  ;;  ;;  ;;  ;; Lưu địa chỉ DATA vào AX  ;; DS = AX  ;; AH = 9  ;; DX = OFFSET STR1  ;; In ra màn hình STR1  ;; Gọi Input  ;; Lưu BX vào Stack  ;;  ;;  ;;  ;;  ;; Lấy đỉnh Stack lưu vào CX  ;; BX = BX + CX  ;; X = BX  ;;  ;;  ;; In ra màn hình STR3  ;; Gọi PRINT\_ENTER  ;; Hàm xử lí Fibonacci  ;; AX = X  ;; AX = AX – 1  ;; X = AX  ;; Gọi S1\_TO\_S3  ;;  ;;  ;;  ;;  ;; CX = X  ;; CX = CX + 1  ;; CX = CX – 1. Lặp FIBO đến khi CX=0  ;; Nhảy tới END\_PROGRAM  ;;  ;;  ;;  ;; BX = 0  ;;  ;; AH = 1  ;; Nhập số từ bàn phím  ;;AL = 13 thì cờ ZF = 1. Hai dòng này có nghĩa khi nhấn Enter thì sẽ kết thúc nhập  ;;AL = AL – 30H  ;;AH = 0  ;; Lưu AX vào Stack  ;; AX = 10  ;; AX = AX x BX  ;; BX = AX  ;; Lấy đỉnh Stack lưu vào AX  ;; BX = BX + AX  ;;  ;;  ;;  ;;  ;;  ;;  Procedure thực hiện lưu các phần tử của mảng S1 vào mảng S3  ;;  Procedure thực hiện lưu các phần tử của mảng S3 vào mảng S2  ;;  Procedure thực hiện tính tổng các phần tử của mảng S1 với S2 theo chỉ số tương ứng. Kết quả được lưu vào mảng S1  ;;  Procedure thực hiện in các phần tử của mảng S1  ;;  Xuống dòng  ;;  ;; Kết thúc chương trình |