|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **4. COMPUTE NCR USING RECURSIVE PROCEDURE ASSUMETHAT NAND R ARE NON-NEGATIVE INTEGERS** | | | | | | |
| **LABEL** | **NEMONIC** | **OPERAND (MEMORY/REGISTER)** | | |  | **COMMENTS** |
|  | **.MODEL** | **SMALL** | | | **;** | **ONE DATA & CODE SEGMENT** |
|  | **.DATA** |  | | | **;** |  |
| **N** | **DW** | **04** | | | **;** | **N VALUE** |
| **R** | **DW** | **02** | | | **;** | **R VALUE** |
| **NCR** | **DW** | **00** | | | **;** | **NCR VALUE** |
|  | **.CODE** |  |  |  | **;** |  |
|  | **MOV** | **AX** | **,** | **@DATA** | **;** | **INITIALIZATION OF** |
|  | **MOV** | **DS** | **,** | **AX** | **;** | **DATA SEGMENT** |
|  | **MOV** | **AX** | **,** | **N** | **;** | **COPY N TO AX** |
|  | **MOV** | **BX** | **,** | **R** | **;** | **COPY R TO BX** |
|  | **CALL** | **NCRPROC** |  |  | **;** | **CALL RECURSIVE NCR=(N-1)CR + (N-1)C(R-1)** |
|  | **INT** | **3** |  |  | **;** | **CONTROL BREAKPOINT** |
|  |  |  |  |  | **;** |  |
| **NCRPROC** | **PROC** |  | **;** |  | **;** |  |
|  | **CMP** | **BX** | **,** | **AX** | **;** | **IF R=N THEN NCR=1** |
|  | **JE** | **RES1** |  |  | **;** | **JUMP TO UPDATE NCR** |
|  |  |  |  |  | **;** |  |
|  | **CMP** | **BX** | **,** | **01** | **;** | **IF R=1 THEN NCR=N** |
|  | **JE** | **RESN** |  |  | **;** | **JUMP TO UPDATE NCR** |
|  |  |  |  |  | **;** |  |
|  | **DEC** | **AX** |  |  | **;** | **MAKE N TO (N-1)** |
|  |  |  |  |  | **;** |  |
|  | **CMP** | **BX** | **,** | **AX** | **;** | **IF R=N-1 THEN NCR=N** |
|  | **JE** | **RESN1** |  |  | **;** | **JUMP TO UPDATE NCR** |
|  |  |  |  |  | **;** |  |
|  | **PUSH** | **AX** |  |  | **;** | **SAVE N AND R** |
|  | **PUSH** | **BX** |  |  | **;** |  |
|  | **CALL** | **NCRPROC** |  |  | **;** | **CALL FOR (N-1)CR** |
|  | **POP** | **BX** |  |  | **;** |  |
|  | **POP** | **AX** |  |  | **;** | **RESTORE N AND R** |
|  |  |  |  |  | **;** |  |
|  | **DEC** | **BX** |  |  | **;** | **MAKE R TO (R-1)** |
|  |  |  |  |  | **;** |  |
|  | **PUSH** | **AX** |  |  | **;** | **SAVE (N-1) AND (R-1)** |
|  | **PUSH** | **BX** |  |  | **;** |  |
|  | **CALL** | **NCRPROC** |  |  | **;** | **CALL FOR (N-1)C(R-1)** |
|  | **POP** | **BX** |  |  | **;** |  |
|  | **POP** | **AX** |  |  | **;** | **RESTORE (N-1) AND (R-1)** |
|  |  |  |  |  | **;** |  |
|  | **RET** |  |  |  | **;** | **RETURN** |
| **RES1:** | **INC** | **NCR** |  |  | **;** | **ADD 1 TO NCR FOR R=N** |
|  |  |  |  |  | **;** |  |
|  | **RET** |  |  |  | **;** | **RETURN** |
| **RESN1:** | **INC** | **NCR** |  |  | **;** | **INCREMENT NCR** |
| **RESN:** | **ADD** | **NCR** | **,** | **AX** | **;** | **AND ADD N-1 TO NCR** |
|  | **RET** |  |  |  | **;** | **FOR R=N-1 & R=1 RETURN** |
| **NCRPROC** | **ENDP** |  |  |  | **;** | **END OF PROCEDURE** |
|  | **END** |  |  |  | **;** | **END OF PROGRAM** |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **10. DRIVE A STEPPER MOTOR INTERFACE TO ROTATE THE MOTOR IN SPECIFIED DIRECTION (CLOCKWISE OR COUNTER-CLOCKWISE) BY N STEPS (DIRECTION AND N ARE SPECIFIED BY THE EXAMINER). INTRODUCE SUITABLE DELAY BETWEEN SUCCESSIVE STEPS. (ANY ARBITRARY VALUE FOR THE DELAY MAY BE ASSUMED BY THE STUDENT).** | | | | | | |
| **LABEL** | **NEMONIC** | **OPERAND (MEMORY/REGISTER)** | | |  | **COMMENTS** |
|  | **.MODEL** | **SMALL** | | | **;** |  |
|  | **.DATA** |  | | | **;** |  |
| **PA** | **EQU** | **0XXXXH** | | | **;** | **ADDRESS OF PORT A** |
| **PB** | **EQU** | **0XXXXH** | | | **;** | **ADDRESS OF PORT B** |
| **PC** | **EQU** | **0XXXXH** | | | **;** | **ADDRESS OF PORT C** |
| **CTRL** | **EQU** | **0XXXXH** | | | **;** | **ADDRESS OF CONTROL WORD** |
|  | **.CODE** |  | | |  |  |
|  | **MOV** | **AL** | **,** | **80H** | **;** | **INITIALIZATION OF 8255 CONTROL WORD** |
|  | **MOV** | **DX** | **,** | **CTRL** | **;** |  |
|  | **OUT** | **DX** | **,** | **AL** | **;** |  |
|  | **XOR** | **AH** | **,** | **AH** |  | **CLEAR AH** |
|  | **MOV** | **CX** | **,** | **50** |  | **NUMBER OF STEPS** |
|  | **MOV** | **AL** | **,** | **33H** |  | **INITIALIZE CONTOL BIT(0011) LS 4 BITS** |
|  |  |  |  |  |  |  |
|  | **MOV** | **DX** | **,** | **PC** |  | **SEND CONTROL BIT VIA PORT C** |
| **BACK:** | **OUT** | **DX** | **,** | **AL** |  |  |
|  | **CALL** | **DELAY** |  |  |  |  |
|  | **ROL** | **AL** | **,** | **01H** |  | **TO GENERATE NEXT STEP** |
|  | **LOOP** | **BACK** |  |  |  | **LOOP UNTIL CX=0** |
|  |  |  |  |  |  |  |
|  | **INT3** |  |  |  |  |  |
|  |  |  |  |  |  |  |
| **DELAY** | **PROC** |  |  |  | **;** | **DELAY PROCEDURE TO ELAPSE TIME** |
|  | **PUSH** | **CX** |  |  |  | **SAVE THE CONTENTS OF CX** |
|  |  |  |  |  |  |  |
|  | **MOV** | **CX** | **,** | **0FH** | **;** | **SET OUTER LOOP COUNT** |
| **OUTER:** | **PUSH** | **CX** |  |  | **;** |  |
|  |  |  |  |  | **;** |  |
|  | **MOV** | **CX** |  | **0FFFFH** |  | **SET INNER LOOP COUNT** |
| **INNER:** | **NOP** |  |  |  | **;** | **NO OPERATION INSTRUCTIONS** |
|  | **NOP** |  |  |  | **;** |  |
|  | **NOP** |  |  |  | **;** |  |
|  | **LOOP** | **INNER** |  |  | **;** | **GOTO INNER** |
|  |  |  |  |  |  |  |
|  | **POP** | **CX** |  |  | **;** |  |
|  | **LOOP** | **OUTER** |  |  | **;** | **GOTO OUTER** |
|  |  |  |  |  |  |  |
|  | **POP** | **CX** |  |  |  | **RESTORE THE CONTENTS OF CX** |
|  |  |  |  |  |  |  |
|  | **RET** |  |  |  | **;** | **RETUREN TO CALLING PROCEDURE** |
| **DELAY** | **ENDP** |  |  |  | **;** | **END OF PROCEDURE** |
|  |  |  |  |  |  |  |
|  | **END** |  |  |  | **;** | **END OF PROGRAM** |