DATA GENERAL CORPORATION

Maintenance Familiarization Course

Study Assignment No. 1

A. Reading Assignment:

1. Read pages 2-1 through 2-7, 2-10 through 2-17, and pages 2-21 (section 2.3) through 2-25 in the How to Use the Nova Computers.

2. Briefly look over pages 2-25 through 2-29, and pages 2-52 through 2-56 in How to Use the Nova Computers.

B. Solve for the values in all four accumulators and carry, when the following program halts.

Initial Conditions: AC0 = 146004
AC1 = 047632
AC2 - 063417
AC3 = 014321

C = 1

Location	Octal Instructions
500 /	000412
501 /	174400
502 /	135300
503 /	117520
504 /	142304
505 /	135400
506 /	010405
507 /	000777
510 /	134400
511 /	063077
512 /	000767
513 /	177776

(Be careful; this program jumps around!)

FOR THIS PROBLEM, ASSUME THAT ALL FOUR ACCUMULATORS CAN HANDLE 4 BITS ONLY* WITH A 1-BIT CARRY.

FIND THE VALUE OF ACCUMULATORS AC0 & AC1 AT HALT.

GIVEN: (BINARY)
AC0 = 0101
AC1 = 0101
AC2 = 0011
AC3 = 0000

C = 0

520/177774
177760

START → 476/004402

INSTRUCTIONS ARE 16 BIT WORDS

477/063077

500/102460

501/054003

502/034416

503/125203

504/101201

505/143220

506/175404

507/000774

510/125260

511/002003

* EXCEPT FOR AC3 WHICH WILL BE LARGE ENOUGH TO CONTAIN A 9-BIT ADDRESS, AND THE 16-BIT WORD 177774.

SOLUTION TO 4-BIT ACCUMULATOR PROBLEM

```
476/ 004402 = JSR 500                    520/ 177774
477/ 063077 = HALT
500/ 102460 = SUB C 0,0
501/ 054003 = STA 3,3   ← [STORE CONTENTS OF AC3 AT LOCATION 000003]
502/ 034416 = LDA 3,.+16
503/ 125203 = MOVR 1,1,SNC
504/ 101201 = MOVR 0,0,SKP
505/ 143220 = ADDZR 2,0
506/ 175404 = INC 3,3,SZR
507/ 000774 = JMP .-4
510/ 125260 = MOVCR 1,1
511/ 006003 = JMP @3
```

INITIALLY

AC0 = 0101
AC1 = 0101 C=0
AC2 = 0011
AC3 = 0000

ANALYSIS

476/ JSR 500

AT THE END OF THIS INSTRUCTION, AC3 WILL CONTAIN THE RETURN ADDRESS (477) [REMEMBER, AC3 IS LARGE ENOUGH TO CONTAIN A 5-BIT ADDRESS.

500/ SUB C 0,0 CARRY IS COMPLIMENTED FIRST (=1). WHEN AC0 IS ADDED TO THE TWO'S COMPLIMENT OF AC0, AN OVERFLOW WILL OCCUR, AND CARRY WILL BE COMPLIMENTED AGAIN.

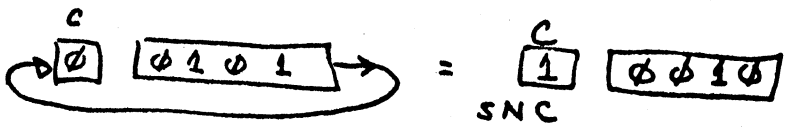
AC0 = 0000, C = 0 AT COMPLETION OF 500.

501/ STA 3,3 STORE CONTENTS OF AC3 IN LOCATION 3, SINCE WE WISH TO USE AC3 IN THIS SUBROUTINE.

502/ LDA 3,.+16 LOAD THE CONTENTS OF LOCATION 520 IN AC3. AC3 = 177774 AT COMPLETION.

503/ MOVR 1,1,SNC

C=1 AC1=0010



505/ ADDZR 2,0 C=0 0011 C=1 AC0=0001

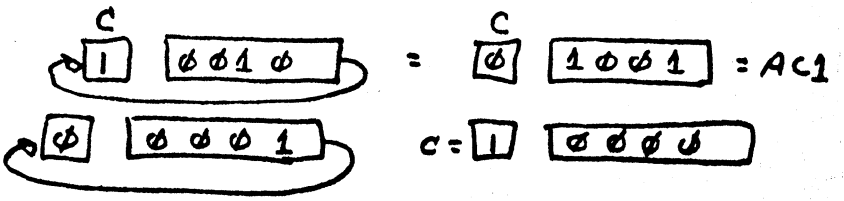
506/ INC 3,3,SZR AC3 WILL EQUAL 177775 (NO SKP)

507/ JMP 503

503/ MOVR 1,1,SNC

504/ MOVR 0,0,SKP

C=1 AC0=0000



THE MULTIPLICAND OR ZERO. HENCE, AT EACH STEP WE EITHER ADD THE MULTIPLICAND AND SHIFT OR SIMPLY SHIFT DEPENDING ON WHETHER THE NEXT BIT OF THE MULTIPLIER IS 1 OR Ø.

THIS SUBROUTINE OPERATES ON UNSIGNED INTEGERS IN AC1 AND AC2 TO GENERATE A DOUBLE LENGTH PRODUCT WHOSE HIGH AND LOW ORDER PARTS ARE STORED IN ACØ AND AC1.

FOR THE FOUR-BIT ACCUMULATOR PROBLEM,

AC1 IS MULTIPLIED BY AC2 (Ø1Ø1 × ØØ11)

THE HIGH ORDER RESULT IS STORED IN ACØ = ØØØØ

THE LOW ORDER RESULT IS STORED IN AC1 = 1111

(SINCE Ø1Ø1 × ØØ11 = 1111)

THIS PROGRAM (SUBROUTINE) CAN EASILY BE MODIFIED TO HANDLE A 16-BIT ACCUMULATOR. SIMPLY CHANGE THE CONTENTS OF LOCATION 520 TO READ -

"177760" [TWO'S COMPLIMENT OF 20]

THIS MODIFICATION ADJUSTS FOR ALL 16 BITS.

^C0 = 146004.
AC 1 = 047632.
AC 2 = 063417.
AC 3 = 014321.

C = 1.

500/ JMP .+12
501/ NEG 3,3.
502/ MOV,S, 1.3.
503/ AND,Z,R, 0,3.
504/ ADC,S,2,0,SZR.
505/ INC 1,3.
506/ ISZ, 1 .+5
507/ JMP .1 .377 → JMP .1-1
510/ NEG 1,3.
511/ HALT
512/ JMP .-11
513/ 177776.

What Program Does.

500/ Jump to 512.
512/ Jump to 501.
501/ AC3 goes to 163457 C=1
502/ AC3 goes to 115117 C=1
503/ AC3 goes to 104004 C=0

504/ AC0 goes to 172144 c = 0
505/ AC3 goes to 047633 c = 0
506/ 513 now contains 177777
507/ Jump back one address.
506/ 513 now contains 0. skip to 510
510/ AC3 contains 130146.
511/ HALT.

RESULT.

AC0 = 172144.
AC1 = 047632.
AC2 = 063417.
AC3 = 130146.

c = 0.

ALC Program Famliarization.

AC 0 . 000002
AC 1 . 143241
AC 2 . 000777
AC 3 . 001000

C = 0.

500 / 162300 . ADC 5.3.0
501 / 143200 ADD R.2.0
502 / 162400 SUB 3.0
503 / 154400 NEG 2.3
504 / 162400 SUB 3.0
505 / 063077 HALT.

500 / 176777
000002

177001

AC0 = 000776 C=0

501 / 000776
000777

001775

AC0 : 000776 C=1.

502/ 176777 1's compl.

+1

177000 2's Compl.

000776

177776

AC0 = 177776 c = 1.

503/ 177000

+1

177001

AC3 = 177001.

504/ 177001

000776

+1

000777

177776

000775

AC0 = 000775 c = 0.

AC0 = 000775

AC1 = 143241

AC2 = 000777

AC3 = 177001

DATA GENERAL CORPORATION

NOVA 1200 MAINTENANCE FAMILIARIZATION COURSE

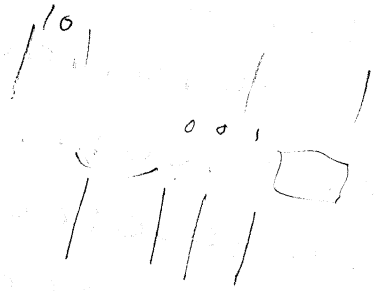
STUDY ASSIGNMENT No. 2

A. READING ASSIGNMENT

1. READ PAGES 2-55 AND 2-56 AND 2-21 THROUGH 2-29 IN How to Use the Nova Computer.

2. LOOK OVER PAGES C1 THROUGH C8 IN THE NOVA ~~1220~~ 840 TECHNICAL MANUAL. C5 C7

B. WRITE A SHORT PROGRAM THAT WILL OUTPUT YOUR NAME USING A TELETYPE. THE REQUIRED ASCII CODE IS ON THE BACK OF YOUR PROGRAM CARD.



— NIOS.
~ SKPDN
— JMP. 0 — 1
— — GO TO LOAD.
— DOAS
— 152 - 550
— JMP. - 5.
— HALT

177774

TROUBLESHOOTING PHILOSOPHY

Effective trouble shooting is accomplished in a minimum of time by following a series of logical steps. The ultimate aim is to effectively pinpoint the actual problem using all information available. Locating the malfunction is the next logical step. The following is a suggested plan for effective error analysis:

a) Investigation - record the state of the machine on error occurrence. Look for obvious symtons including operator error, loose plugs or connectors, blown fuses or tripped circuit breaker.

b) Isolation - through the use of diagnostic programs or console trouble shooting techniques, attempt to isolate the malfunction to a particular board.

c) Component Isolation - Isolate the faulty component using an oscilloscope and short diagnostic loops either toggled in at the console or as part of a diagnostic. Select the correct external synch.

d) Replace the faulty component and retest by running the diagnostic that originally failed.

e) Record for future reference, the symptoms, cause, and unique trouble shooting methods used to isolate the malfunction.

```
/                    NOVA 1200 LOGIC TEST

/1.     ABSTRACT
/           NOVA 1200 LOGIC TEST IS A MAINTENANCE PROGRAM
/           DESIGNED TO TEST THE NOVA 1200 CENTRAL PROCESS-
/           ING UNIT. IT IS A GATE BY GATE TEST OF THE
/           LOGIC USED TO IMPLEMENT THE NOVA 1200 INSTRUCT-
/           ION SET. THE TEST DOES NOT INCLUDE ANY INPUT
/           OR OUTPUT EQUIPMENT.

/2.     MACHINE REQUIREMENTS
/2.1        NOVA 1200 PROCESSOR
/2.2        4K READ/WRITE MEMORY

/3.     SWITCH SETTINGS
/3.1        STARTING ADDRESS=400

/4.     OPERATING PROCEEDURE
/4.1        LOAD THE PROGRAM VIA THE BINARY LOADER
/4.2        SET THE SWITCHES TO 000400
/4.3        PRESS START
/4.4        PROCESSOR SHOULD HALT
/4.5        PRESS CONTINUE

/5.     ERROR DESCRIPTION
/5.1        THE HALT INSTRUCTION IS USED TO INDICATE ERRORS.
/5.2        WHEN A ERROR IS DETECTED RECORD THE STATE OF THE
/           MACHINE. CONSULT THE LISTING FOR POSSIABLE
/           CAUSES OF FAILURE. CONSTRUCT A PROGRAM LOOP
/           WHICH WILL REPRODUCE THE ERROR. SCOPE THE LOGIC.

/6.     PROGRAM DESCRIPTION
/           THIS PROGRAM IS A COLLECTION OF SMALL ROUTINES
/           EACH DESIGNED TO TEST A PORTION OF THE PROCESSOR
/           LOGIC. EACH ROUTINE IS DESIGNED TO TEST AS
/           SMALL A PART OF THE LOGIC AS POSSIABLE. EACH TEST
/           IN THE SEQUENCE IS BASED ON PREVIOUS TEST WORKING.

/7.     MISC
/           THE TIME FOR ONE COMPLETE PASS IS MEASURED IN
/           MILLISECONDS.


.EOT
```

```
00644 176005  A47:    ADC 3,3,SNR       ;ADD COMPLIMENT ANY NUMBER
00645 063077          HALT              ;TO ITSELF SHOULD PRODUCE
                                        ;MINUS ONE. CHECK INPUT
                                        ;TO SKIP LOGIC AND/OR GATE
                                        ;WITH MBC13.

00646 176400  A48:    SUB 3,3           ;ZERO RESULT SHOULD NOT
00647 175043          MOVO 3,3,SNC      ;AFFECT STATE OF CARRY SKIP.
00650 063077          HALT              ;CHECK AND/OR GATE TO SKIP LOGIC.

00651 176000  A49:    ADC 3,3           ;ADC INSTRUCTION PRODUCES 177777
00652 175005          MOV 3,3,SNR       ;IN AC3. IF IR6 OF THT "MOV"
00653 063077          HALT              ;FAILS THE RESULT IN AC3 WILL
                                        ;BE ZERO AND NO SKIP.

00654 020122  A50:    LDA 0,K0          ;AC0 SHOULD BE LOADED WITH
00655 024123          LDA 1,M1          ;ZERO AND INCREMENTED TO
00656 101422          INCZ 0,0,SZC      ;+1. IF THE LOAD AC1 INSTRUCTION
00657 063077          HALT              ;LOADED ZERO ALSO THE
                                        ;INC WOULD SET THE CARRY.
                                        ;CHECK ACS 2 SEL LEVEL.
                                        ;SYNC SCOPE ON LDA AND PACK.

00660 020122  A51:    LDA 0,K0          ;IF LOAD AC2 INSTRUCTION LOADED
00661 030123          LDA 2,M1          ;AC ZERO INSTEAD, THEN INC
00662 101422          INCZ 0,0,SZC      ;INSTRUCTION WOULD SET THE CARRRY.
00663 063077          HALT              ;CHECK ACS 1 SELECT LEVEL
                                        ;DURING THE SECOND "LDA" INSTRUCTION.
                                        ;SYNC SCOPE ON LDA AND PACK.

00664 024122  A52:    LDA 1,K0          ;IF LOAD AC0 REALLY LOADED AC1
00665 020123          LDA 0,M1          ;THE INC INSTRUCTION WOULD
00666 125422          INCZ 1,1,SZC      ;SET THE CARRY. CHECK AC
00667 063077          HALT              ;SOURCE SELECTION FOR LDA.

00670 030122  A53:    LDA 2,K0          ;IF LOAD AC0 REALLY LOADED AC2
00671 020123          LDA 0,M1          ;THE INC INSTRUCTION WOULD
00672 151422          INCZ 2,2,SZC      ;SET THE CARRY. CHECK AC
00673 063077          HALT              ;SOURCE SELECTION FOR LDA.

00674 020130  A54:    LDA 0,K100K       ;AC0 SHOULD CONTAIN 1000000
00675 101005          MOV 0,0,SNR       ;AND THE MOVE INSTRUCTION SHOULD
00676 063077          HALT              ;SKIP. CHECK THE SHIFT 0 INPUT
                                        ;TO THE 4 INPUT OR GATE IN
                                        ;THE ZERO SKIP LOGIC.
```