

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

<u>Location</u>	<u>Octal Instructions</u>
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.

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

C = 0

520/177774

177760

START →

476 / 004402

477 / 003077

500 / 102400

501 / 054003

502 / 034410

503 / 125203

504 / 101201

505 / 143220

506 / 175404

507 / 000774

510 / 125200

511 / 002003

INSTRUCTIONS ARE
16 BIT WORDS

* 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 / 003077 = HALT	
500 / 102400 = SUB C 0, 0	
501 / 054003 = STA 3, 3	← [STORE CONTENTS OF AC3 AT LOCATION 000003]
502 / 034410 = LDA 3, +10	
503 / 125203 = MOVR 1, 1, SNC	
504 / 101201 = MOVR 0, 0, SKP	
505 / 143220 = ADDER 2, 0	INITIALLY
506 / 175404 = INC 3, 3, SER	AC0 = 0101
507 / 000774 = JMP -4	AC1 = 0101 C = 0
510 / 125200 = MOVR 1, 1	AC2 = 0011
511 / 002003 = JMP @ 3	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 9-BIT ADDRESS.]

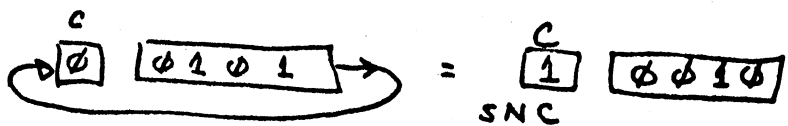
500 / SUB C 0, 0 CARRY IS COMPLEMENTED FIRST (= 1). WHEN AC0 IS ADDED TO THE TWO'S COMPLEMENT OF AC0, AN OVERFLOW WILL OCCUR, AND CARRY WILL BE COMPLEMENTED AGAIN.
 AC0 = 0000, C = 0 AT COMPLETION OF 500.

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

502 / LDA 3, +10 LOAD THE CONTENTS OF LOCATION 520 IN AC3. AC3 = 177774 AT COMPLETION.

503 / MOVR 1, 1, SNC

C = 1 AC1 = 0010

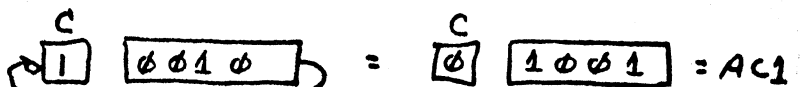


505 / ADDER 2, 0 C = 0 0011 C = 1 AC0 = 0001

506 / INC 3, 3, SER 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 0.

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 AC0 AND AC1.

FOR THE FOUR-BIT ACCUMULATOR PROBLEM,

AC1 IS MULTIPLIED BY AC2 (0101×0011)

THE HIGH ORDER RESULT IS STORED IN AC0 = 0000

THE LOW ORDER RESULT IS STORED IN AC1 = 1111

(SINCE $0101 \times 0011 = 1111$)

THIS PROGRAM (SUBROUTINE) CAN EASILY BE MODIFIED TO HANDLE A 16-BIT ACCUMULATOR. SIMPLY CHANGE THE CONTENTS OF LOCATION S20 TO READ -

" 177760 " [TWO'S COMPLIMENT OF 20]
THIS MODIFICATION ADJUSTS FOR ALL 16 BITS.

$^{\wedge}20 = 146004.$

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, S2R.

505/ INC 1, 3.

506/ ISZ, 1, +5

507/ JMP. 1, 377 \rightarrow JMP. 1-1

510/ NEG 1, 3.

511/ HALT

512/ JMP. -11

513/ 177776.

What Program Does.

500/ Jump to 512.

~~500~~
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/ ACO 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.

ACO = 172144.

AC1 = 047632.

AC2 = 063417.

AC3 = 130146.

C = 0.

ALC PROGRAM FAMILIARIZATION.

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/

$$\begin{array}{r} 176777 \\ + 1 \\ \hline 177000 \\ 000776 \\ \hline 177776 \end{array}$$

1's compl.

2's Compl.

$$Ac_0 = 177776 \quad c = 1.$$

503/

$$\begin{array}{r} 177000 \\ + 1 \\ \hline 177001 \end{array}$$

$$Ac_3 = 177001.$$

504/

$$\begin{array}{r} 177001 \\ 000776 \\ + 1 \\ 000777 \\ 177776 \\ \hline 000775 \end{array}$$

$$Ac_0 = 000775 \quad c = 0.$$

$$Ac_0 = 000775$$

$$Ac_1 = 143241$$

$$Ac_2 = 000777$$

$$Ac_3 = 177001$$

DATA GENERAL CORPORATION

NOVA 1200 MAINTENANCE FAMILIARIZATION COURSE

STUDY ASSIGNMENT No. 2

A. READING ASSIGNMENT

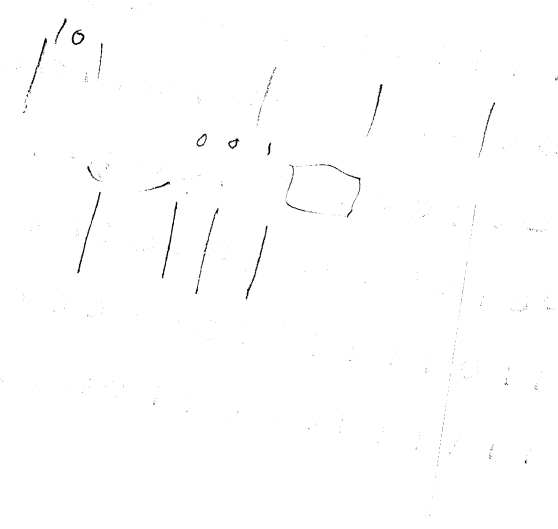
1. READ PAGES 2-55 AND 2-56 AND 2-21 THROUGH 2-29 IN
HOW TO USE THE NOVA COMPUTER.
2. LOOK OVER PAGES ~~C1~~ THROUGH ~~C6~~ IN THE NOVA ~~1220~~ ~~810~~
TECHNICAL MANUAL. ~~C5~~ ~~C7~~

- 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.

C

[Faint, illegible handwritten notes]

[Faint, illegible handwritten notes]



— NIOS.

— SKPDN

— JMP. 0-1

— Go TO LOND.

— DOAS

— 152-558

— JMP. -5.

— HALT

177774

C

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 symptoms 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
11. ABSTRACT
1 NOVA 1200 LOGIC TEST IS A MAINTENANCE PROGRAM
1 DESIGNED TO TEST THE NOVA 1200 CENTRAL PROCESS-
1 ING UNIT. IT IS A GATE BY GATE TEST OF THE
1 LOGIC USED TO IMPLEMENT THE NOVA 1200 INSTRUCT-
1 ION SET. THE TEST DOES NOT INCLUDE ANY INPUT
1 OR OUTPUT EQUIPMENT.
12. MACHINE REQUIREMENTS
12.1 NOVA 1200 PROCESSOR
12.2 4K READ/WRITE MEMORY
13. SWITCH SETTINGS
13.1 STARTING ADDRESS=400
14. OPERATING PROCEEDURE
14.1 LOAD THE PROGRAM VIA THE BINARY LOADER
14.2 SET THE SWITCHES TO 000400
14.3 PRESS START
14.4 PROCESSOR SHOULD HALT
14.5 PRESS CONTINUE
15. ERROR DESCRIPTION
15.1 THE HALT INSTRUCTION IS USED TO INDICATE ERRORS.
15.2 WHEN A ERROR IS DETECTED RECORD THE STATE OF THE
1 MACHINE. CONSULT THE LISTING FOR POSSIBLE
1 CAUSES OF FAILURE. CONSTRUCT A PROGRAM LOOP
1 WHICH WILL REPRODUCE THE ERROR. SCOPE THE LOGIC.
16. PROGRAM DESCRIPTION
1 THIS PROGRAM IS A COLLECTION OF SMALL ROUTINES
1 EACH DESIGNED TO TEST A PORTION OF THE PROCESSOR
1 LOGIC. EACH ROUTINE IS DESIGNED TO TEST AS
1 SMALL A PART OF THE LOGIC AS POSSIBLE. EACH TEST
1 IN THE SEQUENCE IS BASED ON PREVIOUS TEST WORKING.
17. MISC
1 THE TIME FOR ONE COMPLETE PASS IS MEASURED IN
1 MILLISECONDS.

.EOT

