ECE441 MONITOR DESIGN PROJECT
SPRING 2016

Design Objective:

We have learned the different aspects of hardware design such as memory and I/O design with a MC68000 processor. The goal of this Design Project is to build up a Monitor program with MC68000 assembly language. Your itor program should be able to perform basic debugger functions (e.g., Memory Display, Memory Sort, Memory Modify, Block Fill, Block Search, Block Move), and ability to handle exceptions (e.g., bus, address, illegal instruction, etc.). There are four components to fulfil the requirement of this Design Project:

1. Command Interpreter

The Monitor program recognizes which command is entered, and branches to subroutine(s) for executing this command. If there is no match, it displays an error message. You must include adequate error handling and error checks in the Command Interpreter.

2. Debugger Commands

You are required to implement *all 12 commands described in Page 2 & 3, and 2 additional commands* of your own for the full Monitor implementation. You must provide the following for each command that you implement:

- Full description of the functionality for each command
- Detailed explanation of the command usage
- Sample output for each command

For your 2 additional commands, you must define appropriate commands and provide adequate descriptions with the basis for your selection of the commands and minimum of 3 references. Your Monitor program must be able to execute your selection of commands after the prompt (e.g. "MONITOR441>") and terminated with a carriage return <CR>. The group of commands must deal with memory operations including memory display, modify, move, search and memory testing.

3. Exception Handlers

Your Monitor program must be able to handle ***ALL*** 8 system exceptions of MC68000:

- Bus & Address Error Exception **(must include SSW, BA, IR Outputs. Refer Lab 3 Experiment)**
- Illegal Instruction Exception
- Privilege Violation Exception
- Divide by Zero Exception
- Check Instruction Exception
- Line A & Line F Emulator Exception

*** For those who are using EASy68K, you must enable "**Enable Exception**" option in SIM68K before running*

4. **User Instruction Manual**

   Your Monitor program should include a help screen for every command and their syntax such as the required parameters and operands.

Project Details:

Debugger Commands

Your Monitor program must be able to execute the following commands after the prompt "MONITOR441>" and terminated with carriage return, <CR>.

You MUST use the command names as specified below. You should come up with your own syntax based on the command specification. If any specific instruction is mentioned in the command description, you must follow and fulfil the requirement.

HELP (Help)

- The Help command must display all available commands and usage descriptions.

MDSP (Memory Display)

- The MDSP (Memory Display) command outputs the address and memory contents from <address1> to <address2>.
- The MDSP (Memory Display) command **ALSO** outputs the address and memory contents from <address1> to <address1 + 16bytes>.

SORTW (Sort)

- The SORT command sorts a block of memory. The starting address <address1> and the ending address <address2> of the memory block are specified in the command. The order (A or D) specifies whether the list is sorted in Ascending or Descending order. (The size of the data to be sorted is a word)

MM (Memory Modify)

- MM (Memory Modify) command is used to display memory and, as required, modify data or enter new data. The size (byte, word, long word) controls the number of bytes displayed for each address.

MS (Memory Set)

- The Memory Set (MS) command alters memory by setting data into the address specified. The data can take the form of ASCII string or hexadecimal data.

BF (Block Fill)

- The Block Fill (BF) command fills memory starting with the word boundary <address1> through <address2>. Both <address1> and <address2> must be even addresses. This command only fills with a word-size (2 bytes) data pattern. If an entire word-size data pattern is not entered, the pattern is right justified and leading zeros are inserted.

BMOV (Block Move)

- The Block Move (BMOV) command is used to move (duplicate) blocks of memory from one area to another.

BTST (Block Test)

- The Block Test (BT) command is a destructive test of a block of memory beginning at <address1> through <address2>. If this test runs to completion without detecting errors, and display a message that no error was detected.
- If memory problems are found, a message is displayed indicating the address, the data stored, and the data read of the failing memory.

BSCH (Block Search)

- The BSCH (Block Search) command is used to search a literal string in a memory block starting at <address1> through <address2> both inclusive. In BSCH command, if search finds matching data, the **data** and **address(es)** must be displayed.

GO (Execute Program)

- The GO command is used to start execution from a given address.

DF (Display Formatted Registers)

- The Display Formatted Registers (DF) command is used to display the MC68000 processor registers. This command should display current PC, SR, US, SS and D, A registers. *Note that since your program will be using D and A registers, those changed values should not be shown to the user. Thus, you must save all D and A register values before any of your subroutines*.

EXIT (Exit Monitor Program)

- The EXIT command terminates/exits your Monitor program.

- In addition to above 12 commands, you MUST define ***2 additional commands*** for this project. You are required to implement and include detailed descriptions for these 2 additional commands, as well as for the required 12 commands.

- *Commands that has similar functionality will NOT be treated as separate commands, but considered as one command.*

ECE441 MONITOR DESIGN PROJECT
SPRING 2016

Design Constraints:

The program and look up tables have the following **practical design constraints**:

- Entire code must be smaller than 3K size starting from $1000 (including look-up tables for help and error messages).
- 1 K stack size residing in memory locations $3000 and up
- Include any relevant I/O Trap #14(for SANPER) or #15(for EASy68K) routines in code
- ***YOU MAY NOT USE MACROS AT ALL IN THIS PROJECT! You will get ZERO POINTS if found.***
- Erroneous inputs should not kill program but the number of errors statements should be minimum.

Requirements:

- ***This project should be finished individually and independently***. However, you are encouraged to discuss with your classmates.
  - You must acknowledge all of the work including figures, codes and writings are belonging to you with your signature on the front page of your Project Report.
  - **If any similarity in the code, comments, customized program behavior, report writings and/or figures are found, both the helper (original work) and the requestor (duplicated/modified work) will be called for academic disciplinary action, failure of this course, and expulsion from IIT ECE Department.**
  - IIT Code of Academic Honesty: https://web.iit.edu/student-affairs/handbook/fine-print/code-academic-honesty
- Your program has to be command driven and properly organized.
- Your program code has to be demonstrated on the SANPER-1 ELU or EASy68K individually to the TA.
- Instructions for each command must be fully described in the project report. Error messages must guide the user for correct usage of the commands.
- **A Project Report must be submitted in hard copy (WILL NOT accept any electronic copies)**

Grading Policy:

- Program Demo (Points: 200)
  - TA will evaluate your program using the "ECE441 Design Project Evaluation Form" on Page 6:
    - Testing and evaluation of the command interpreter
    - Functional evaluation of each command
    - Evaluation of exception handling
    - Clarity of presentation, syntax error handling and help messages
  - *Demonstrate whatever you did and show your work. Without any demo, your report will not be accepted nor graded*
  - *Any modification of the command name is not acceptable*

- ***If any similarity in the code, comments, and/or customized program behavior are found, both the helper (original work) and the requestor (duplicated/modified work) will be called for academic disciplinary action, failure of this course, and expulsion from IIT ECE Department.***

- Final Report (Points: 200)
  - Your final report will be evaluated using the "ECE441 Design Project Evaluation Form" on Page 7 :
    - Table of contents and Abstract
    - Clear and concise description of the implementation of commands
    - A separate section for the flowcharts/algorithms used in
      - Command Interpreter
      - Individual Commands
      - Exception Handlers
    - Code listings with global and local comments
    - A quick manual for command usage and monitor
    - A separate section where you highlight the engineering and design challenges and methods to overcome these challenges
    - A separate section discussing the expansion of the existing Monitor program for more advanced implementations. How would this project help if you are developing an OS for an embedded system using MC68000? Provide an example and discuss the features for this.
    - Conclusion and References
  - ***If find any similarity in figures, flowcharts, and/or writings are found, both the helper (original work) and the requestor (duplicated/modified work) will be called for academic disciplinary action, failure of this course, and expulsion from IIT ECE Department.***

Program Code Due Date & Demo Time

- **Section 1 : Wednesday, April 27th 2016, 1:50PM-4:30PM**
- **Section 2 : Wednesday, April 27th 2016, 6:25PM-9:05PM**
- You must submit your program code to TAs during your program demo
  - Individual demo schedule will be determined in mid-April

Report Due Date

- **TBD**
  - No Electronic Submissions, Late Submissions will **NOT** be accepted nor graded

ECE441 DESIGN PROJECT EVALUATION FORM

Student Name:

Date:

Project Demonstration Evaluation Score:

Command Interpreter	**Command 1 – 7**						
	HELP	MDSP	SORTW	MM	MS	BF	BMOV
No Credit							
Partial Credit							
Full Credit							
Command Interpreter	**Command 8 – 14**						
	BTST	BSCH	GO	DF	EXIT		
No Credit							
Partial Credit							
Full Credit							

Debugger Commands	**Command 1 – 7**						
	HELP	MDSP	SORTW	MM	MS	BF	BMOV
No Credit							
Partial Credit							
Full Credit							
Debugger Commands	**Command 8 – 14**						
	BTST	BSCH	GO	DF	EXIT		
No Credit							
Partial Credit							
Full Credit							

Exception Handlers	**Exceptions**							
	ADDR	BUS	ILL INS	PRIV	DIV 0	CHK	LINE A	LINE F
No Credit								
Partial Credit								
Full Credit								

ECE441 DESIGN PROJECT EVALUATION FORM

Student Name:

Date:

Project Report Evaluation Score:

SECTIONS		NO CREDIT	PARTIAL CREDIT	FULL CREDIT
English spelling and Grammar	10			
Table of contents	10			
Abstract	10			
Implementation of commands	20			
Flowchart for command interpreter	30			
Flowchart for individual commands	30			
Code listings with comments	20			
A quick user manual	10			
Discussion for engineering and design challenges	30			
Suggestions for advancing your monitor program	10			
Conclusion	10			
References	10			

Grade:

Report	/200
Demo	/200
Total	/400

Note: Demonstration is mandatory. It will take place in the last week of classes. Without any demo, final report will not be accepted nor graded.