Last Revised: Dec1999

0 Objectives

With the completion of this lab, you should be able to:

- Find relevant information in the MUN-88 Manual
- Connect to the MUN-88 using Kermit
- Explain the MUN-88 memory map and its mirror images
- Enter commands in the MUN-88 Monitor
- Enter and run machine code via Monitor commands

1 Introduction

The MUN-88 single board computer will be the focus of all six labs in Engineering 4862. Later, you will be required to write and assemble a number of short programs, and interface several devices. Before you can do these tasks, you must become familiar with the MUN-88. This lab introduces you to the MUN-88, and some of its capabilities. The lab contains a few exercises that introduce you to the monitor commands in MUN-88. Feel free to try out other commands.

It is assumed that you are familiar with the use of the PC and a text editor. For the remaining labs, bring a formatted, 3.5inch floppy disk to store your files (as a backup to your account).

2 The MUN-88 Manual

The ***MUN-88 Single-Board Computer*** manual has two parts. It is important that you understand the layout of the manual, so that you can quickly find the relevant information for future labs.

Part 1 of the manual, the MUN-88 C*ircuit Description*, contains information about the hardware of the system. It begins with a set of 9 sheets with circuit diagrams and part descriptions, and follows with 8 unnumbered pages of detailed comments on hardware design features.

Part 2 of the manual is the *Monitor Reference Manual*, and has 36 numbered pages. It describes the functionality of the MUN-88 Monitor, including available commands and function calls.

3 Using the MUN-88 Monitor

Before you start:

All the DIP switches on the MUN-88 board should be in the closed position before you hit the Reset button. This switch setting represents the baud rate (communications speed) upon reset, and must match the speed of the associated serial port on the PC.

Make sure that the power supply is set to the correct voltages for the board (+5VDC, +12VDC, –12VDC), and that the cables are hooked up correctly. If the board does not respond correctly when you first try to connect, check the power supply. You may need to turn the supply off and on again.

3-1 Kermit

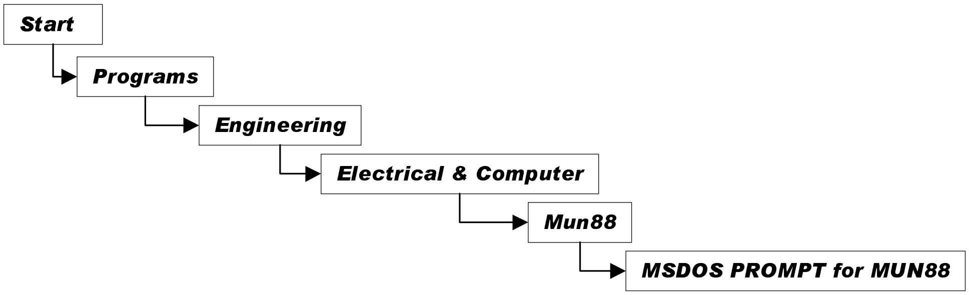
Kermit is a communications program that runs on a PC that can transmit information over the serial link to an external device. In all of these labs, you will use Kermit extensively to communicate with the MUN-88, and to control it via the Monitor environment. When you successfully establish a connection, it will seem as if the keyboard and monitor of the PC is attached to the MUN-88 system.

There are three steps to establish a connection to the MUN-88 via Kermit:

1. Launch MS-DOS from Microsoft Windows
2. Run Kermit
3. Connect to the MUN-88 Monitor environment

<u>Step 1</u>

At the Microsoft Windows desktop, open an MS-DOS window by using the following sequence:



Start → ***Programs*** → ***Engineering*** → ***Electrical & Computer*** → ***Mun88*** → ***MSDOS PROMPT for MUN88***

In addition to providing an MS-DOS prompt, this procedure will mount all necessary files for working with the MUN-88, including Kermit, the assembler, and the linker. In later labs, you would now change to the directories in which your programs were stored.

<u>Step 2</u>

To enter the Kermit environment, type the following at the MS-DOS prompt (only type the words in bold):

```
M:> kermit
```

You will enter the Kermit environment, and see the following prompt:

```
MS-kermit>
```

You may now enter commands to the Kermit environment, but you are not yet connected to the MUN-88. Type ‘**?**’ to see a list of the available commands, or type ’**`help`**’ for more detailed information (follow the on-screen instructions).

Step 3

To connect to the MUN-88, run the provided connection macro:

```
MS-kermit> take mun88.krm
```

This command properly defines the backspace key, connects to the MUN-88, and transfers control. If the MUN-88 board is powered up and the serial cable properly connected to the PC, pressing ***ENTER*** on the PC keyboard will print the MUN-88 prompt on the screen. The prompt in the MUN-88 environment is a period. Once control is transferred to the MUN-88, the Kermit commands are no longer available to you. If you press the ***RESET*** push-button on the MUN-88, you will see the following welcome message on your screen, followed by the period prompt.

Having trouble connecting?

- The power supply should be on.
- The DIP switches should all be off (open).
- The serial cable between the PC and the MUN-88 should be firmly connected.
- Switch S2 should be in RUN position.
- The connection settings on the PC should be 4800 and COM2. Press **Ctrl] C**, and type **take mun88.krm** again.

```
MUN-88 Monitor
Copyright (C) 1988 by David Skoll and Joseph Liang
Free memory begins at 0032:0000
.
```

You are now in MUN-88 Monitor environment, and may interact with the MUN-88 using its Monitor commands. To transfer control back to the PC, you should press the ***Control*** key and the left square bracket key together, and then type in ***C***. That is, ***Ctrl] C***. This will return you to Kermit environment. However, this will not affect your work on the MUN-88. For example, if you enter a program on the MUN-88 and return to Kermit, the program will remain in the memory of the MUN-88.

Once you have successfully connected to the MUN-88, you need only type **`c`** (or **`connect`**) at the Kermit prompt to reconnect to the MUN-88. However, this is only true if you do not leave Kermit and return to MS-DOS.

To return to the MS-DOS environment from Kermit, type

```
MS-kermit> exit
```

or

```
MS-kermit> quit
```

3-2 Some Monitor Commands

Reconnect to the MUN-88 through Kermit (you will have to type **`take mun88.krm`** again at the Kermit prompt). At the dot-prompt, type **`help`** to view a list of MUN-88 commands available to you. More detailed information on each command is available in Part II of the MUN-88 Manual. You will notice a standard format for the presentation of these commands. Please consult the manual for some of the finer points of entering commands, but two of the more important points are:

- You need only type the capitalized letters of commands – such as **`m`** for **`modify`**.
- Anything listed between ‘<’ and ‘>’ is optional – thus for modify, you do not need to enter a value.

3-3 Memory Map

From the listing of commands on the screen, determine the commands to read and write to the memory of the MUN-88.

In your MUN-88 Manual, there is a memory map of the system. Using this memory map, and the information that was presented to you in the MUN-88 welcome message, identify the RAM area (address range) available for the user code/data.

For a particular memory address, say 005E8 H, devise 4 different segment:offset pairs. Confirm that your pairs are equivalent by using the above commands.

MUN-88 Monitor Code

Where does the monitor code reside in this system? Try changing the contents of any of these locations. Explain why you were or were not successful.

Mirror Images

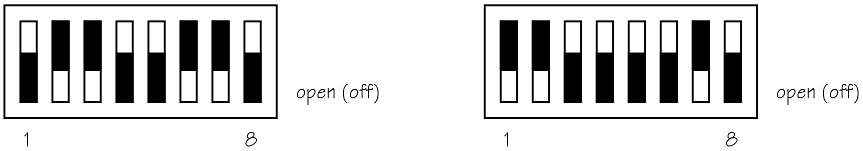
Note that there are several mirror images of the same memory locations due to partial decoding of the 8088’s memory addressing lines. Look at *sheet 2* of the MUN-88 circuit diagrams in the manual. This sheet shows the connection of the RAM, ROM, and switch input port. The U10 device is the RAM decoder (with 8 output ports, only one of which is shown), and U14 is the 2K SRAM chip. How many address lines are used to select a particular memory element on the SRAM device?

The design choice to have unused address lines give rise to mirror images of memory locations. Why would the designers choose not to use all 20 address lines?

From the number of unused address lines, determine the number of mirror images. List all of the memory addresses that correspond to the 2K SRAM chip. Confirm that the MUN-88 does indeed contain mirror images by using the appropriate monitor commands, and comment.

Input / Output

What Monitor commands read and write data on the I/O ports? Using the input port address corresponding to the DIP switches, read the values for various switch positions. What values do the following patterns give (the white is the switch)?



Using the output port address of the LEDs, write various values and thus change their display. What patterns do 1E and 00 give? Repeat these values with the LED command – what patterns do you see now?

What is the speed of communication (baud rate) between the MUN-88 board and the PC? How can this be changed on the MUN-88?

Programming

Table 0-0 contains a simple program. Using the monitor command **Modify**, key the hexcode into the RAM area allocated for user data and code. Note that the <value> parameter of the modify command is optional.

Try out the following commands: **DReg**, **Reg**, and **Go**. What does this program do? Try out the Single step and Breakpoint commands, and briefly describe how these commands can be used.

Address	Hexcode	Label	Inst	Arguments	Comments
0032:0000	B9 08 00	start:	mov	cx, 8	Eight repetitions
0032:0003	B0 FE		mov	al, 11111110b	Only right-most bit
0032:0005	E6 30	do_it:	out	30h, al	Light one led
0032:0007	D0 C0		rol	al, 1	Next led to left
0032:0009	E8 04 00		call	delay_10ms	Wait 10ms
0032:000C	E2 F7		loop	do_it	Repeat until done
0032:000E	CD 06		int	6	Back to monitor
0032:0010	BA 00 20	delay_10ms:	mov	dx, 2000h	Count approx. 10ms
0032:0013	4A	rep1:	dec	dx	Decrement by 1
0032:0014	75 FD		jnz	rep1	If not done, repeat
0032:0016	C3		ret		Over

Table 0-0

Submission

At the end of the lab session, submit your answers to all the questions, and include comments where appropriate. Include your pre-lab.