TO: Drs. Victor P. Nelson and John Y. Hung
FROM: "Uncovered Warriors," Stephen Taylor, and Brian Arnberg (Section 3)
DATE: August 26, 2011
SUBJECT: Project Creation and Debugging

The primary objective of the first lab session was to familiarize each student with the development process for programming the Freescale MC9S12C32 in the *CodeWarrior* Development studio. The primary objective was achieved by having each team program the chip with two pre-written programs. The secondary objective of the lab was for each student to begin practicing and using the communications skills discussed in the lecture. The secondary objective was achieved by having each student begin maintaining his engineering notebook and by having him write a two-page memo concerning the lab.

The two programs used to test the development software were written in assembly and C. Before a team could test the software, though, it had to prepare the chip and board for both the download of the program and the testing. Setting up the board was done per the explanation in the lab manual. The programs were downloaded to the chip via the P&E USB BDM Multilink module. Once the board and chip were set up, *CodeWarrior* was set up per the instructions in the lab manual. Brian Arnberg's terminal would not correctly load the software, so Stephen Taylor's had to be used. This would not be worth mentioning, if it hadn't taken five minutes for the computer to login. After *CodeWarrior* was set up, each team tested the assembly code given in the manual. It had two errors, one on line 35 ("ldx" was written "lx") and one on line 37 (the variable "Inner" had been typed "Iner"). Once the code was corrected, it compiled correctly and downloaded successfully to the chip. Once each team was satisfied with the assembly code, it went on to test the C code, which was without error. Even though the C code was without error, Brian had incorrectly copied part of the code, thus there was an issue with the compilation. Once the typo that Brian had produced was found, it was easily corrected. Once the typo was corrected, the code correctly compiled and downloaded to the chip. As with the assembly code, the C code functioned as it was described in the lab manual. Because of the work done in the lab, each student now has a better understanding of the process used to develop code for the Freescale MC9S12C32, debug the code, and properly download the code to the chip. Therefore, the primary objective was achieved.

Beginning to maintain the engineering notebook was not very difficult. As each team stepped through the lab manual, each student made notes in his notebook concerning what they had done. For instance, each student noted which lines had errors with it, what the issues were, and how they were fixed. Each team might have also made note of any issues they had while they were setting *CodeWarrior* up. Additional notes, like the amount of time it took to complete certain steps, might also have been recorded. Once each student submits his memo, if he has also started writing in his notebook, the secondary objective will be complete.

Both of the programs used did the same thing: They waited for input from a push button so that the state of an LED could be toggled. The code accomplished this by polling the first bit, PE0, until it was 1. When it was one, there was a delay of about a second, and the state of the LED was toggled by complementing the state of the LED bit, PA0. The C code was more understandable, so it is a good thing that the rest of the programming done for the semester will be done in C.

Code development in *CodeWarrior* was not the only thing that students learned about in class. For instance, Brian learned that a Windows 7 machine cannot read a USB storage device that has been formatted as ext3 and that he would have to bring a storage device formatted as fat32 or ntfs. Considering that both of the objectives were met, it would have to be concluded that this was a successful lab.