Memo to: Randy Larimer
From: Johnny Gaddis
Date: 3/30/17
Regarding: EELE 465, Lab 4

Summary:

The purpose of lab 4 was to use the HCS908QG8 to store a value for serial communications and then send it back through serial communications. This was accomplished by waiting for a receive flag to be set in the SCI status register and then storing the value received in RAM. The program continued to do this until a carriage return or hex value $0d was read. Once this was read, all the values read in to RAM were printed back out followed by a new line. Once this routine completed it starts again for more repetition. This lab led to a greater understanding of the HCS908QG8 while showing students practical uses of serial communications with microcontrollers! The program uses as much RAM as input characters entered, in the program summary there are 69 bytes of read only memory and 65 bytes of read/write memory.

Setup:

The setup required was rather minimal. We needed to put our microcontroller back on to the demo board and plug in a SCI cord. The code was setup to initialize both receiving and transmitting with no interrupts.

SCI Solution:

Firstly, a subroutine was needed to wait for a receive flag. Once this flag was set the value was stored in RAM offset by X register, then X was incremented. This looped until enter (a carriage return) was hit. When enter was hit, PuTTy was setup to force a new line and the values were transmitted back to the computer. I initially tried to use interrupts to achieve the lab goals. This ended up being not as clean and easy as a flag clearing approach.

Final Thoughts:

This lab taught me how to use serial communications. There was some research required for this lab such as going through the HCS908QG8 manual and reading the serial communication chapter. The work required to set up the lab was very minimal. This SCI feature of the microcontroller showed me additional practical uses of our microcontroller.

Appendix A Flowcharts:

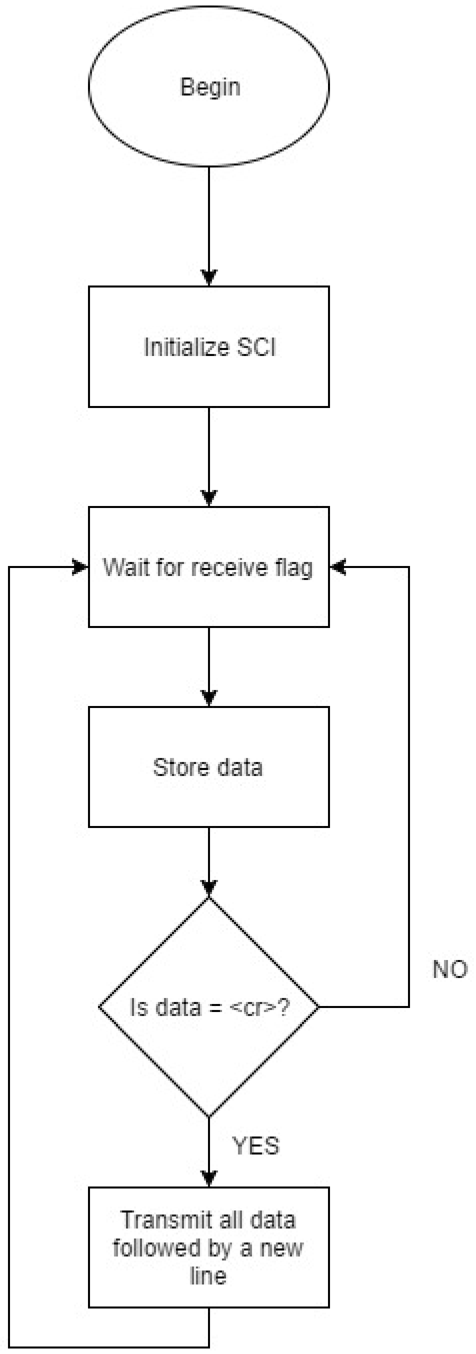

Figure 1: Interrupt Routine and Program Flow