COE4DS4 – Lab #5 Report
Group Number: 05
Febuary 13, 2018

Exercise 1:

We replaced in the ISR counter expired to Real Madrid. The time each printf took was approximately 0.005 ms to complete. From the waveform (jtag uart av irq) the actual start of the interrupt is 6.36663ms and completion time is 6.45361 ms. In total interrupt service routine time was 6.45361-6.392= 0.08698 ms. The first letter R was printed at 6.397 ms while the last letter was printed at 6.448 ms.

Exercise 2:

In this exercise we find the longest strictly monotonic decreasing subsequence from random data written to a DP RAM. We find this sequence through hardware and software to verify the data. Different offsets are set to either read and write data. Offset 0 we read the value in the DP RAM at the specified address. Offset 1 we write a value to the specified address. Offset 2 we read the address of the starting value of the subsequence. Offset 3 holds the length of the subsequence. Offset 4 when going from 0 to 1 begins the sequence search. Offset 5 clears the interrupt. We first use Offset 1 to write values randomly to the DP RAM. Then we use Offset 0 to check if values were properly written. Offset 4 is then used to begin the search for the sequence which triggers an interrupt using Offset 5. The interrupt then prints the values read from Offset 2 and 3. After this is completed in hardware we also check the values in software NIOS using an algorithm in NIOS similar to the one implemented in hardware in our CUSTOM_DEC_UNIT.

Exercise 3:

In this exercise the experiment 3 was modified to manipulate images in BMP format. From the spec bmp sheet, we got made the header for bmp a bmp image. Also from the file we found that the order of the colours changes in bmp file type. In PPM it is RGB while in BMP it is BGR (blue, green, red) this is due to the fact that bmp is in little endian format and the addresses are flipped around. We were tasked to down sample the image to make it into a 320*240 size. In order to down sample the image, in our double for loop we check if we are currently in an even column or row if yes then write BGR values otherwise we skip the odd row and column. To display the image from the SD card to the middle of the LCD on 320 * 240, we check if we are in the correct location if yes then we set the BGR values otherwise we set them to 0 which is black and finally we write the line.