ME 333 Introduction to Mechatronics
Quiz 3

1. The 4 GB virtual memory map is broke into 8 512 MB segments. Of these 8, only 2 are used, and they are called kseg0 and kseg1. What is the difference between these two segments?

kseg0 is cacheable, kseg1 is not cacheable

2. You are manipulating the bits of SFR LATC.
(a) Write one line of code, using LATCCLR, that clears bits 3 and 4 of LATC. (Remember that the bits are numbered 0 to 31.)

LATCCLR = 0x18; or
LATCCLR = 0b11000; or
LATCCLR = 24;

(b) Write one line of code, using LATCSET, that sets bits 2 and 5 of LATC.

LATCSET = 0x24; or
LATCSET = 0b100100; or
LATCSET = 36;

(c) Write one line of code, using LATCINV, that inverts bits 0 and 1 of LATC.

LATCINV = 0x3; or
LATCINV = 0b11; or
LATCINV = 3;

3) Your C code uses the SFR TRISB. It compiles correctly because the header file p32mx795f512l.h is included, which declares TRISB as an `extern unsigned int`. Explain clearly why references to TRISB in your code get resolved to the correct virtual address of the SFR TRISB on the PIC32.

The header file declares the variable. The linker matches TRISB to the SFR virtual address defined in the processor.o file. "Extern" tells the linker not to allocate memory in RAM for TRISB.

4) The bootloader program is the first program to run when your PIC32 is reset, regardless of any program you have installed.
(a) If you are pressing the USER button when the PIC32 is reset, what does the bootloader do?

It attempts to communicate with the bootloader utility (nu32utility) on your computer, and can receive and write .hex files to _RESET_ADDR.

(b) If you are not pressing the USER button at reset, what does the bootloader do?

It goes to program flash and executes the code written there.

5) When you build your program, you use the command of the form
`xc32-gcc —mprocessor=32MX795F512L ...` Explain two ways the flag `—mprocessor=32MX795F512L` is used in the build.

It allows xc.h to function properly by defining the constant __32MX795F512L__.
It tells the compiler which PIC model to target.

6) Your NU32 code has the global variable array definition
`int x[123456];`
Does this generate a compiler error, a linker error, a run-time error, or not of the above? Explain.

This creates and array of length 123456. Each element of the array is an int, which is 32 bits. Therefore this array occupies 123456*32 = 3950592 bits = 482 kB. The memory in the PIC's RAM is 128 kB. Therefore this array is too large. This will generate a linker error.