

CSE3442 Embedded Systems I

Test 1 Example Study Topics

Due to similarity to test questions, no solutions will be provided for these questions.

- 1.** Understand the different ways to control GPIO pins using the read-modify-write on the 32-bit register (method of stop_go.c) and bitmapped control (method of stop_go_bitband.c). Be able to calculate the address that controls a bit in the peripheral register within the bitbanded region (Lab 2).
- 2.** Understand the function waitMicrosecond(us) completely. Make sure that you can both analyze the code and determine the number of clock cycles and the elapsed time for a similar code snippet or function. Be prepared for different system clock rates (Lab 3).
- 3.** Write a function, void wait1ms(void) that can be called from C, which waits 1ms as accurately as possible given a system clock rate of 20 MHz. Implement this code with one loop. Implement this function with less than 30 lines of code.
- 4.** Know how to pass up to 4 parameters to a function and how a result is returned from a function. As a simple example, where would a function, uint32_t sum4(uint16_t a, uint16_t b, uint16_t c, uint16_t d) that returns a+b+c+d, look for the values a-d and how would it return the result?
- 5.** Implement this function above in assembly language. In general, be prepared to use the B, BL, BX, PUSH, POP, AND, OR, ADD, SUB, CBNZ, CBZ, LDR, and STR operations. Also make sure you are prepared to use the byte and half-word modifiers and to use pre- and post-modifiers for data operations.
- 6.** Understand why the assembly code for putsUart0() uses a POP{PC} instead of the usual BX LR to return from the function. Also understand why R4 is pushed and why R4 is used to store the value in R0.
- 7.** Understand the initHw() function completely and be prepared to configure any or all of the 35 I/O pins on the board to be either a digital input/output (AHB or APB) or for an auxiliary function like a UART1.
- 8.** Configure any or the UARTs to operate at 9600baud with a 33.3 MHz clock rate with 8 bits data, no parity and 1 stop bit. Be prepared to solve this problem with a range of options (clock, baud rate, stop bits, data bits).
- 9.** Write a short C routine that will set a green LED connected as on the evaluation kit if the string "unlock" is received in sequence on UART0.
- 10.** Write a short routine that initializes timer 4 to create interrupts at 60Hz (or as close as possible).
- 11.** Understand how frequency and time can be measured with counter(s).

Please make sure you have all relevant pages of the evaluation board manual, datasheets, C calling and register convention document, class notes, and class code printed out before the exam (the sections should be obvious from the class examples, reading assignments, and these test questions). No computers are allowed during the exams as stated in the syllabus. Be sure to bring a calculator.