CSE/EE 5/7385 Microcontroller Architecture and Interfacing

HOMEWORK 4

1. An embedded systems designer is creating an arbitrary waveform generator that will output a periodic waveform based upon a set of data values stored in a lookup table. The designer has created the table that consists of 400 32-bit Q.31 values. The arbitrary waveform is specified to have a frequency of 10 kHz. (NOTE: k is the SI unit of 1000).

a) To meet the 10kHz specification, give the delay value (in units of time) between the output of each value stored in the lookup table.

b) Give an appropriate ARM instruction for loading the waveform value from the lookup table into register **r0** and whose base address is stored in **r1**. Furthermore assume the pointer register is **r2**. The pointer value is an integer that cyclically has incremented values running from 0 to 399 (in decimal); that is, **r2** is the offset from the base address of the lookup table that points to the current waveform value.

c) Assume your answer in part a) above is 500μs (NOTE: this is NOT necessarily the correct answer for part a), it may or may not be). Also assume that the ARM program that outputs the values is of this form:

```
loop: ;load lookup value (1 instruction)
      ;output (str) value to output device (1 instruction)
      ;increment the pointer register by 1 (1 instruction)
      ;if r1 is 400 or greater, set r1 to 0 (3 instructions)
      ;go to loop (1 instruction)
```

If each instruction above requires 1 CPU clock cycle to execute, what is the minimum ARM core clock speed (ie. frequency) required to meet the 10kHz specification for the arbitrary waveform generator?

d) How much memory will the lookup table occupy (in units of KB or MB)?

2) How many external interrupt inputs (electrical inputs) are present on an ARM processor core and what are their names?

3) Give two reasons why the FIQ is faster than the IRQ for processing interrupts.

4) Most interrupt/exception table entries are jumps to another area in memory where the actual exception handler code is present. Why are these exception table entries jump instructions?

5) In addition, to a jump, another type of instruction can be stored in the exception vector table that causes the processor to execute a handler stored elsewhere in the memory. Give an example of this type of instruction and indicate why it might be used instead of a jump.

6) Why might an embedded system designer choose to use "**`lr`**" rather than "**`r14`**" in their source code?