CSCI 4415 Lab # 2

Ringtone Player

Created: January 30, 2011

Updated: January 29, 2012

- Clarified requirements

Objective:

- Make a ringtone player out of the ZNEO Contest Kit.

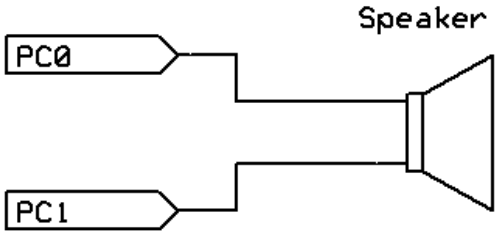
Time Estimate:

- 10 hours.
- You should be prepared to start this after class 4 (GPIO, Interrupts, Basic Timers) and this will be due by Midnight class 7.

Specific Requirements:

Your program will play one of three ringtones when a button is pressed. The ringtones will be stored as a RTTL string in memory. When a button is pressed you will parse the string, extract the necessary data, and then emit the proper musical note and duration on the speaker. As the tune is playing you will show the note on the LED array.

1. Connect the speaker (in your lab kit) to GPIO pins PC0 and PC1. An audio amplifier is not necessary.



2. Write a function to parse RTTL and extract the necessary data (note frequency and duration). For the ringtone format, use RTTL (Nokia Ring Tone Transfer Language).
3. Your main program should parse 3 tunes from RTTL strings and play them when one of the buttons is pressed (tune one for SW1, tune 2 for SW 2, etc). Include the tunes as ASCII strings in your programs and parse the text to play the tunes (don't encode the tune as a sequence a note and timing information). **Do not pre-parse** the RTTL and include data table in your program.
4. You should implement as much RTTL as is necessary to play the three tunes you have selected (probably everything that is listed on the Wikipedia RTTL except dotted notes, you can ignore the dots rather than edit them out of the ringtones).

5. Look up the frequencies for the various notes so your program is reasonably accurate. http://en.wikipedia.org/wiki/Piano_key_frequencies
6. While the tune is playing on the speaker you will also display the note/duration on the LED display (display each note, no need to scroll the stream of text as that may take longer than the note takes to play). You should make reusable libraries for the sub-components of this lab.
7. There are some ringtones on the class webiste (ringtone_samples.txt on the files page) that your program should be able to play without difficulty, but you can include your own tunes if your would rather.
8. Create a file named readme.txt file that includes any notes about the lab and answers to the questions below.
9. Your main() function should be in a file named main.c (any other files can be named as you choose). In the main.c file include the following information in a comment block.

```
Author: [Your name]
Email:  [Your email]
Class:  [CSCI-4415]
Date:
Lab:    [Lab number]
Description: [short description of program. Include changes you made and
note any enhancements that you made to the lab]
Other files: [list other files that are necessary for this program]
Compile: [provide any special instructions for compiling. Only necessary if
there are special instructions]
Problems: [explaining what you did, what problems you had, how you solved them,
and what you might do differently if you had to do it again]
Comments: [Feel free to provide comments on how this lab went, what you think
is good or bad about it and how it could be improved or anything else you want
to say.]
Enhancements: [describe the extra capability you added to this lab]
Answers: [answer the questions below]
```

10. Turn in a zip file (lastname-lab2.zip) of the entire project directory (include the entire ZDSII project directory). Use BlackBoard to submit assignments (or if thats not working for some reason email the ZIP file).

References

1. RTTL is documented here:
    - http://en.wikipedia.org/wiki/Ring_Tone_Transfer_Language
    - http://electronics.howstuffworks.com/ringtone1.htm.
3. Here is asimilar project (just in case you need examples or hints) for a different processor but the same basic idea:
    - http://www.beyondlogic.org/pic/ringtones.htm
4. Lots of free ringtones in various formats here:
    - http://www.2thumbswap.com/members/tones/nokia/tones_nokia_main.html
5. iMelody format:
    - http://en.wikipedia.org/wiki/IMelody
5. Use the ringtone convedrter to find other formats.
    - http://merwin.bespin.org/t4aphp/

Grading

- On time, in proper format (Zip file), named <lastname>-lab1.zip
- Compiles and runs without problems
- Meets lab requirements
- Reasonable code (partitioning of function, coding standard, readable
- Reasonable comments
- Parses RTTL at runtime
- Plays 3 tunes minimum
- Plays reasonably accurate tones (frequency) and complete tunes.
- Did you properly set the oscillator speed
- Proper use of Z16
- Answered questions

Enhancements:

- Prompt user using scrolling LED messages
- Implement another ringtone format such as “Nokia Keypress” or “Composer Code” or “iMelody” some other documented format.

Questions

Include with your lab submission (in the comments of your main.c file):

1. Could you have used other IO pins (if I have not specified). Why/Why not?
2. How did you configure the GPIO pins?
3. What is the highest frequency note you could play? Whats the lowest? What limits the frequencies?
4. How can you verify your notes are accurate (the right frequency)? Did you verify your notes? Were your notes accurate?
5. Did you store your user defined tune in SRAM or FLASH memory? How did you specify that? Why did you choose that? How can you verify that?