Homework #3: Audio Processing

Design a Windows Phone to fulfill the following specifications. Submit a .zip file of all contents of your repository to the class dropbox. To reduce upload time, be sure to "clean" your visual studio project before zipping.

Phase 1: Your windows phone application will communicate with a Arduino in order to send a message to it. The arduino will then translate that message into beeps emitted through the Piezo element attached. The example solution defines a protocol for first detecting the start of an "audio packet", and then transmitting a single bit (see lecture slides for details). Your solution may be more adventurous by transmitting multiple bits per audio packet, etc… Base requirements are to use Frequency Shift-Keying for transmitting data.

Phase 2: Your windows phone application will detect the incoming data and decode it properly roughly 90% of the time. Loud interfering sources may be able to trigger accidental bit reception, this is acceptable. Information should be displayed to the user, including the fourier transform of the received signal (the example solution displays only the relevant data portion of the received signal, which can be helpful during debugging) a visual display of when a signal is detected (In the example solution this is shown as a function of the preamble-tuned matched filter in white) as well as the decoded message.

The example solution uses a pivot to separate the transmit and receive panels. This is atrocious user interface design, and is highly recommended to be improved upon. Additionally, the example solution transmits/receives only 0s and 1s, and only a single bit at a time. This also can be improved, and solutions doing so are more than welcome.