# Heartbeat audio feedback

The heartbeat audio feedback task uses an Arduino to play audio (beep) on each heartbeat. This Arduino is connected to a BIOPAC STP100C digital output and to the stimulus PC through USB (COM port). AcqKnowledge should be set-up so that a pulse is sent to the Arduino on each heartbeat (see “ECG sound Arduino-3.gtl”). From the stimulus PC and the stimulus software one can communicate with the Arduino through the COM port. See “Heartbeat audio feedback setup.png” for a picture of the setup.

## Serial port settings

The Arduino requires the serial port with the following settings: 115200 baud rate, 8 bit, 1 stop bit, No parity (115200, 8N1).

## Protocol

The Arduino is controlled from the stimulus PC by sending a json string to the COM port. This can be done from any desired stimulus presentation software, such as Python or E-Prime. See “ControlHeartBeatAudio.py” for example functions on initializing the serial port, sending a json string and receiving data, and closing the serial port.

Example json string: {"frequency":800, "duration":100, "delay":200,"repeat":10} + Line Feed (\n)

**frequency**: the frequency of the square wave tone in Hz

**duration**: the duration of the tone in ms

**delay**: the delay of the tone with respect to the heartbeat (peak) in ms

**repeat**: the number of times (i.e. heartbeats) the audio feedback should be playing

The Arduino returns the string “Done.” when all audio beeps have been played.

## AcqKnowledge setup

AcqKnowledge should be setup so that it sends a pulse to the Arduino on each heartbeat. This is accomplished by adding a few calculation channels.

* One calculation channel counts the peak. Use the “Rate” Preset, Signal type: HR from ECG (Human Resting) and Function: Count Peaks as output.
* One calculation channel that is high for a duration of 10 ms on each ECG peak. Use the “Expression” preset and enter the following: IF(EQUAL(C0(-20),C0),0,1). In this case, C0 is the previously created channel which counts the peaks.
* Finally, a calculation channel is necessary to output pulses to the Arduino on each peak. Use the “Control” preset. Set source to the previously created channel that is high for each peak. Set output to the digital output channel on which the Arduino is connected. Set Threshold Function Outside 1 with Level (L2) = 0.0 and Level (L1) = 1.1.