



Final Project

T.P.S Report Pt. 2

COVER SHEET

Tad Miller
Danny Nsouli
Systems Programming

Implementation - Heartbeat Monitor

C Code

For this second part of the heartbeat processor, a real time clock has been connected to the arduino which allows a reading to occur every second for more accurate heartbeat data. We repurposed our older command methods so that no functionality is lost, and only added. From a computer, we send a new command to the arduino “visualize”, which allows the arduino to transmit data to the host program to make a histogram. This data is visualized in the terminal’s output using a sequence of characters. The raw values are also wrote to a separate excel file, and can be plotted using an excel histogram.

Arduino Code

For this second part of the heartbeat processor, we added functionality for a real time clock to recorded heartbeat times. A new command ‘v’, or ‘visualize’ was added into the arduino sketch in order to show a visual representation of the histogram. We send from the Arduino to the computer a message which contains the BPM and time stamp. This message of communication utilizes the protocol we developed for this part.

Protocol

For the project we developed a transmission protocol to be used between the C program and the Arduino code. This allows us to send and receive the heartbeat data, as well as the time stamp from the Arduino to our computer. Our protocol is structured in a way that bytes represent each individual section of a message. For instance, the first byte in our message will tell us the message number is coming. We then will see that the next byte after that is the message count (0-255). This repeats for the BPM, and time stamp.

This protocol is developed in such a way that it can be used to reliably send and receive messages from an Arduino to a computer. Because each message is numbered, we can check and see if we are receiving the correct message with each transmission. If something wrong occurs, we can send a retransmission packet from one device to another. This will allow us to get the next message in line consistently.