MEMO: ECE 497

TO: Mark Yoder
FROM: James Savage
DATE: 29 October 2013
SUBJECT: Week 8 Status

---

1. Running `node boneServer.js` creates an HTTP server listening on port 8080 on the Bone, and also initializes a WebSockets server via the Socket.io library. The client side JavaScript, when requested from the Bone via HTTP, can then make a connection back to the bone using the WebSocket. Two WebSockets are actually created, one for writing and one for reading, such that a two-way data channel can be established between the two. The client page can then send updates to the Bone (such as updated row values) and receive updates (such as the current matrix configuration). It is important to note that due to the nature of the JavaScript's architecture, all reads and writes occur asynchronously. This is why reading the matrix involves sending a `matrix` command to the server, and then receiving a `matrix` response at some later point as an indirect product.
2. The client side socket is told to emit an `i2cset` event along with a data payload that sets a new value for that row of the LED Matrix.
3. `.on`
4. The main changes that need to take place is to modify the boneServer.js's `i2cset` event handler to allow all indicies to be set, instead of just every other. The client side can then be modified to also write to all indices of the matrix data array and likewise not discard every other entry in the `matrix` event result data.
5. Yes, I modified `i2cset` to not discard values as noted above.