Higher = more important

Contact Andrew to get more details or clarification

**High-priority tasks**

|  |  |  |
| --- | --- | --- |
| **Task** | **Claim** | **Notes** |
| Basic motor control for showing off | Andrew |  |
| rewrite "getAngle" to work with accelerometer | Rahul |  |
| Rewrite “moveMotor” to work with DC motors | Andrew |  |
| add first order "protocol" function |  | Basically I want a first order function. I'm going to be receiving strings, and depending on the string I receive it activates different functions that actually do stuff.  I want to program the actual functions myself, and add whatever functions I want so you can't hardcode any of the mappings.  This function will handle all commands sent to the simsim over serial.  Also it might be easier if the protocol were rewritten as op-codes (so replace the operation item with an ascii character corresponding to a byte corresponding to an op code or some similar method of encoding, so we have a set number of characters in the string or something). And maybe all the arguments will be integers or characters, instead of a generic string. If an actual ECE/CS major could make the judgement on that... |
| rewrite memory data using proper style |  | Just for good coding practices |
| sleep arduino + BLE + wake up |  | AT commands? |
| populate protocol function |  |  |
|  |  |  |
|  |  |  |

**Less important tasks**

Store position of motor shaft in non volatile memory

Automatically shift motor to get ready for next push to remove startup time.

Determine excessive current that flows for specified period of time and cut off power to motor (prevent motor damage)

Alert users (sound/vision?) about the excessive current (which could be due to excessive load encountered by

motor as the door is improperly aligned with its slot)

Alert users when battery is getting low.

While turning when excessive load is encountered, intelligently increase the motor torque to overcome the excessive load,

then get back to lower torque as SimSim moves the lock to locked/unlocked position.