Demonstration Neural Interface Platform

Owen Leishman
3/8/2024

Motivation

Neural interfaces are devices that interface directly with the human nervous system by using electrical connections to monitor its activity and provide feedback through stimulation. These devices have a wide range of uses in the medical field from enabling patients with paralysis to communicate, to reducing symptoms for people suffering from neurological conditions such as parkinson's disease and epilepsy. These devices are complex and heavily regulated making it hard to learn about them.

This project aims to create a simple neural interface capable of recording and stimulating neurons to demonstrate the basic working principles of such devices. The neural interface will be designed to be as safe as possible but will not meet the safety standards necessary for implantation or FDA approval to simplify the design requirements.

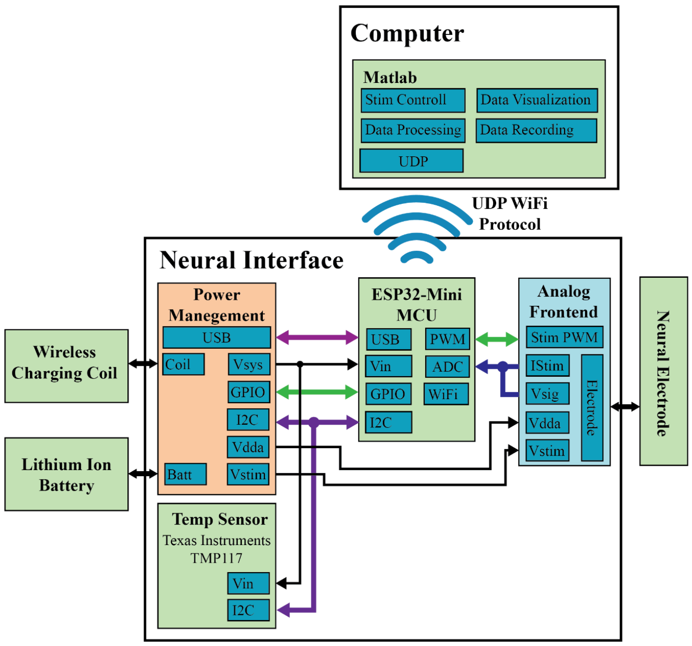
Device Overview

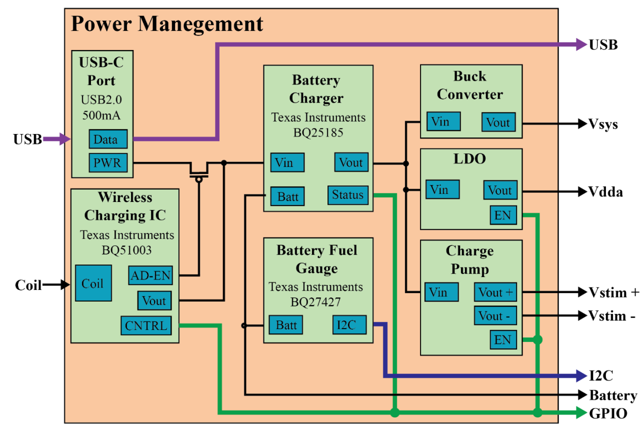
The neural interface will record low level voltages from an electrode (~±0.7V) and transmit its readings over wifi to a computer for visualization and processing and the computer will transmit commands to the neural interface to control its stimulation output. The neural interface will use current controlled stimulation, which applies a constant current through the electrode. Constant current stimulation is desirable over other methods because it deposits a known amount of charge through the electrode. This makes it possible to create a net zero charge in the electrode by alternating positive and negative current stimulations. A buildup of charge in the electrode can be very dangerous as it can cause damage to neurons. The electrode will be made out of a flexible PCB as real neural electrodes are prohibitively expensive for this project. The neural interface will be powered with a lithium-ion battery which can be charged wirelessly or through a USB-C port.

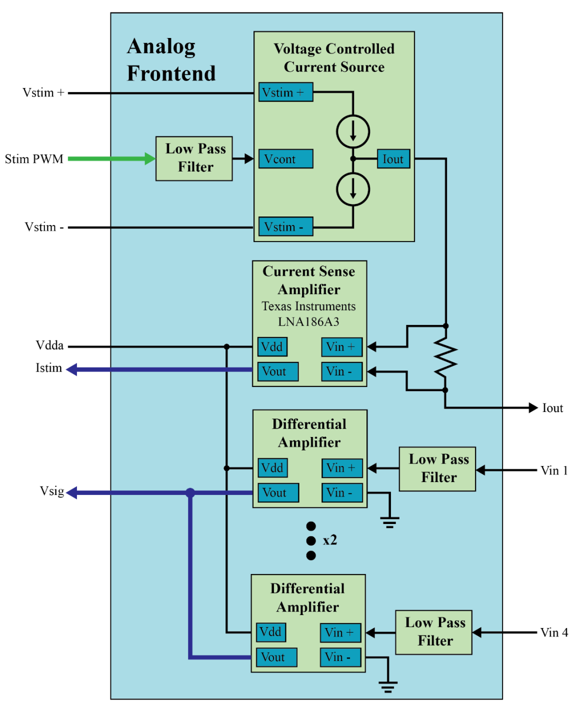
The design of the neural interface is split into three parts, a microcontroller, an analog frontend, and a power management block. An ESP32-S3-Mini was selected for the microcontroller as it is easy to integrate within a system, has an inbuilt wifi transmitter, and contains many peripherals. The ESP32 will use its GPIO pins and an I2C bus to control the power management block. The ESP32 will use PWM to control the stimulation output, and use its ADCs to record electrode voltages from the analog frontend.

The analog frontend will use an enhanced howland current source controlled using PWM to create the stimulation current, with a shunt resistor and current sense amplifier to provide feedback to the microcontroller. The analog frontend also includes amplifiers to shift the differential input voltage (~±0.7V) to a unipolar voltage (0 - 3.3V) that the ESP32's ADCs can read. The constant current source requires a large differential voltage (~±10V) and the amplifier requires a smaller differential voltage (~3.3V) both of which will be provided by the power management block.

The power management block uses a battery charger chip to interface with a lithium-ion battery, this makes it possible to safely charge the battery and prevent the battery from over discharging. A battery fuel gauge is used to allow the microcontroller to easily read how much energy is left in the battery. A wireless charging chip allows the device to be charged using a wireless charger. Various voltage regulator chips are used to create the different voltages needed by the system, these will be chosen according to the noise and power requirements of each power domain. The power management unit is complex, but the wireless charging and fuel gauge components are not necessary for it to work, if the unit proves too difficult the device can be powered externally using a power supply.







Milestones

- Milestone 1: Assemble the device. order components, design and order a PCB, and solder it.
- Milestone 2: Program each subsystem to ensure functionality. Be able to use the ESP32's WiFi, PWM, GPIO, and ADC peripherals. Ensure that the analog front end and power management block work.
- Milestone 3: Integrate all of the subsystems. The ESP32 reads the values from the ADCs and transmits them over WiFi, and can control the stimulation and power management block.