Distributed Chemical Sensing

Michael Caplan, Jacob Rosenstein
Brown School of Engineering

Motivation

- Sensing chemicals in realistic environments is challenging due to complex airflows and low concentrations of target chemical
- Using a distributed network of low-cost chemical sensors enables the system to depend less on the dynamics of the airflow and intelligently combine signals using concepts from information theory

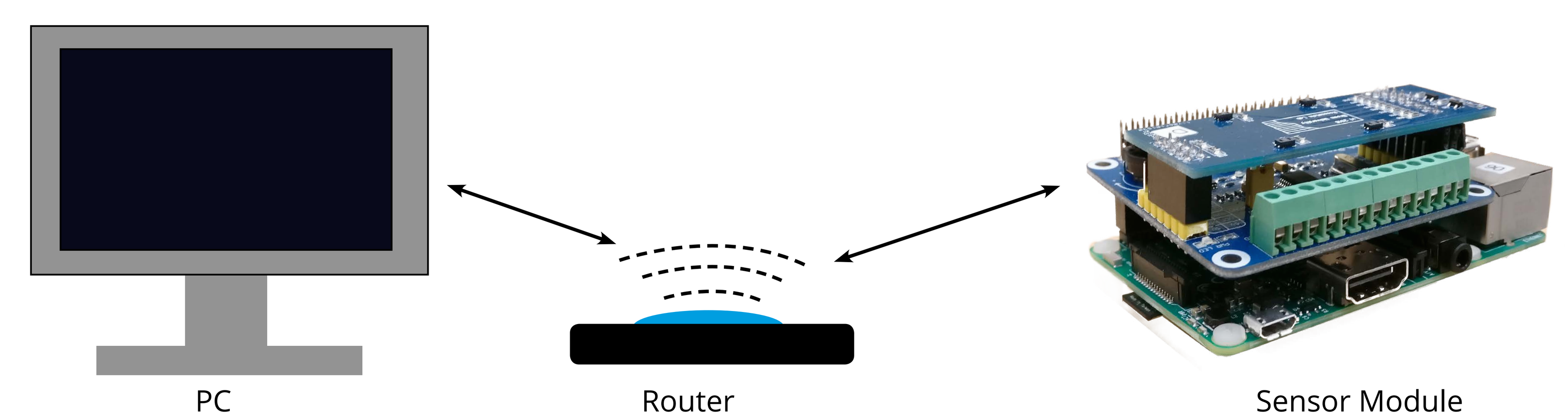
Abstract

- Build small, cheap chemical sensing modules capable of wireless communication
- Build software capable of reading data from multiple software channels as well as saving and sending data
- Create a central program capable of managing multiple units, receiving and saving data for reconstruction and analysis

Progress

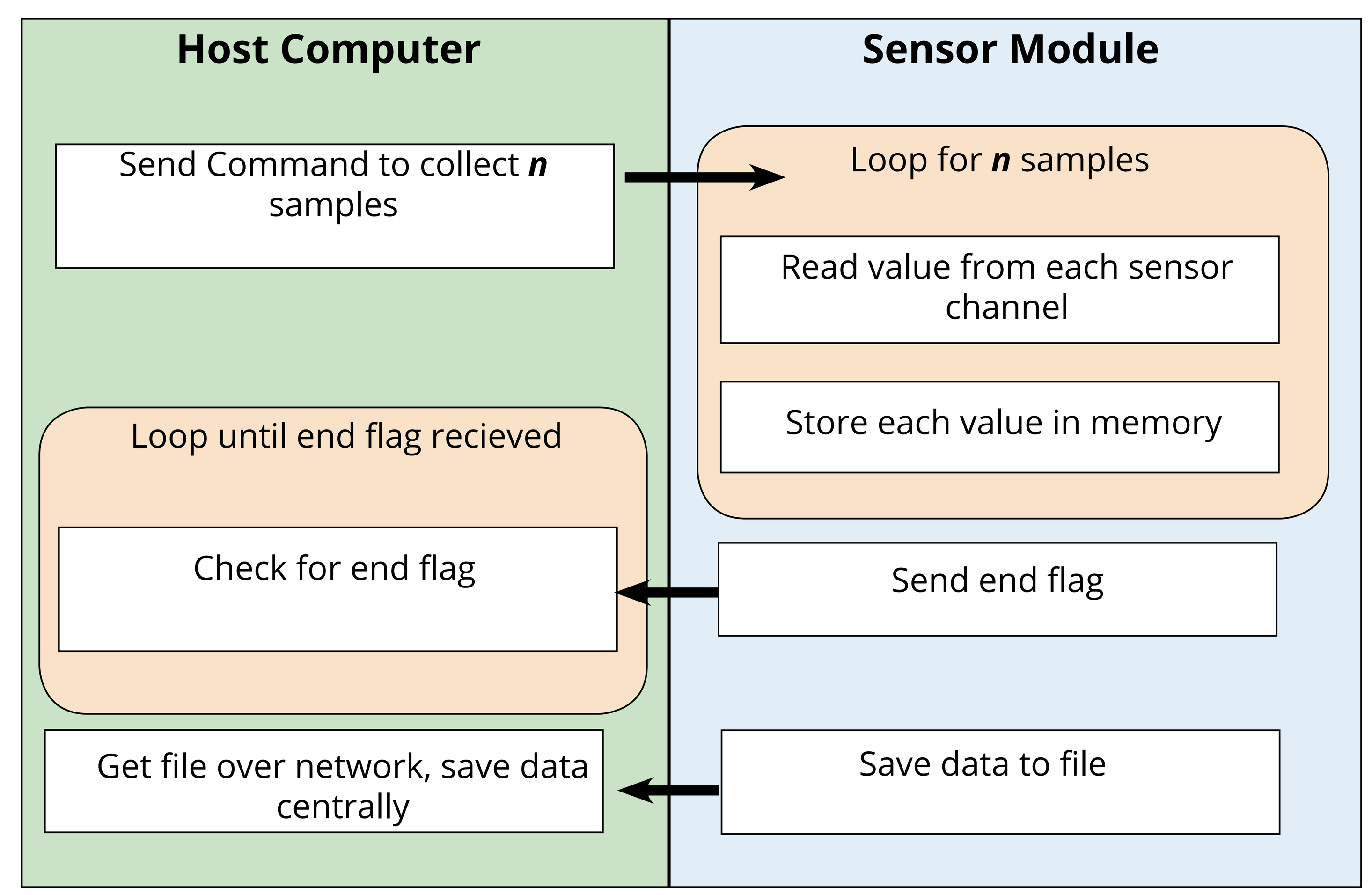
- Changed host to sensor communication from a per-sample UDP broadcast to a method wherein the sensor unit recieves the sample count and sample rate at the trial beginning and transfers data at the trial end
- Changed the data storage from .json to .hdf5, a more rigorous hierarchical file structure that uses space more efficiently
- Previously, the bitrate of the transmit message was bounded by sample rate, because the transmit signal was carried by UDP packets. Now, the message is transferred at the beginning of the trial, and a separate thread is spawned to control the transmitter which allows an arbitrary bitrate
- Designed and laid out a circuit board with ADC and DAC chips, as well as 8 MOX sensors for further trials

Methods



Dimensions: 85 x 56 x 45 (mm)

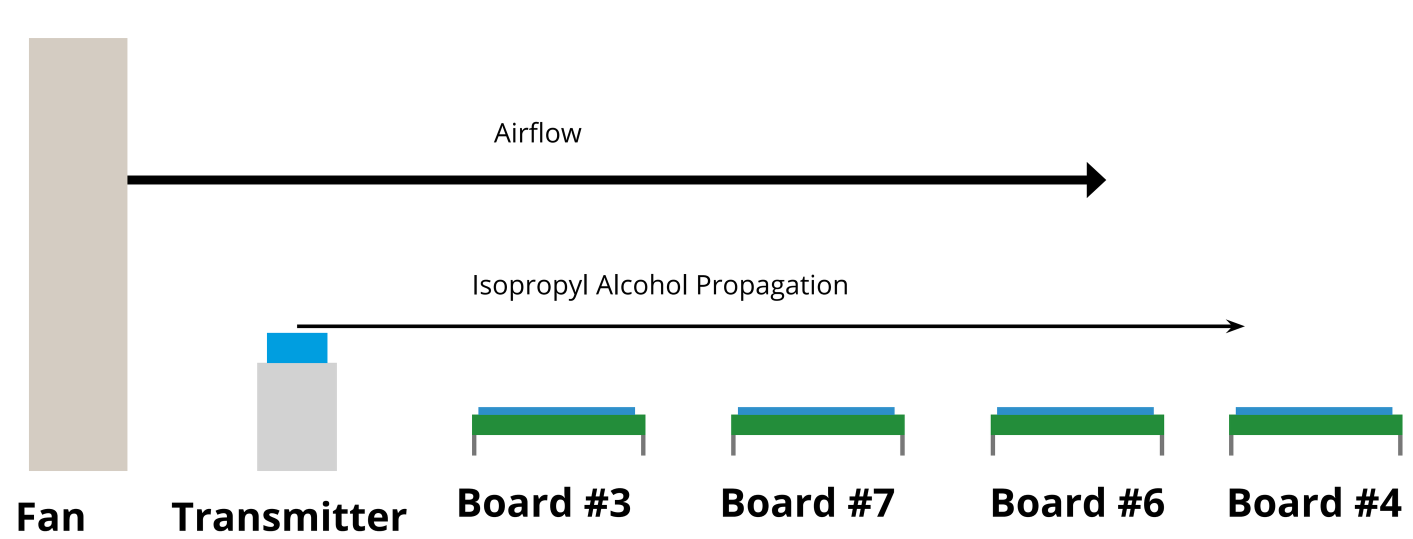
Sensor Module:
- Raspberry Pi
- Analog-Digital & Digital-Analog Converter
- Custom PCB with CCS801 Metal-Oxide Chemical Sensors



Host-Sensor Communication Routine

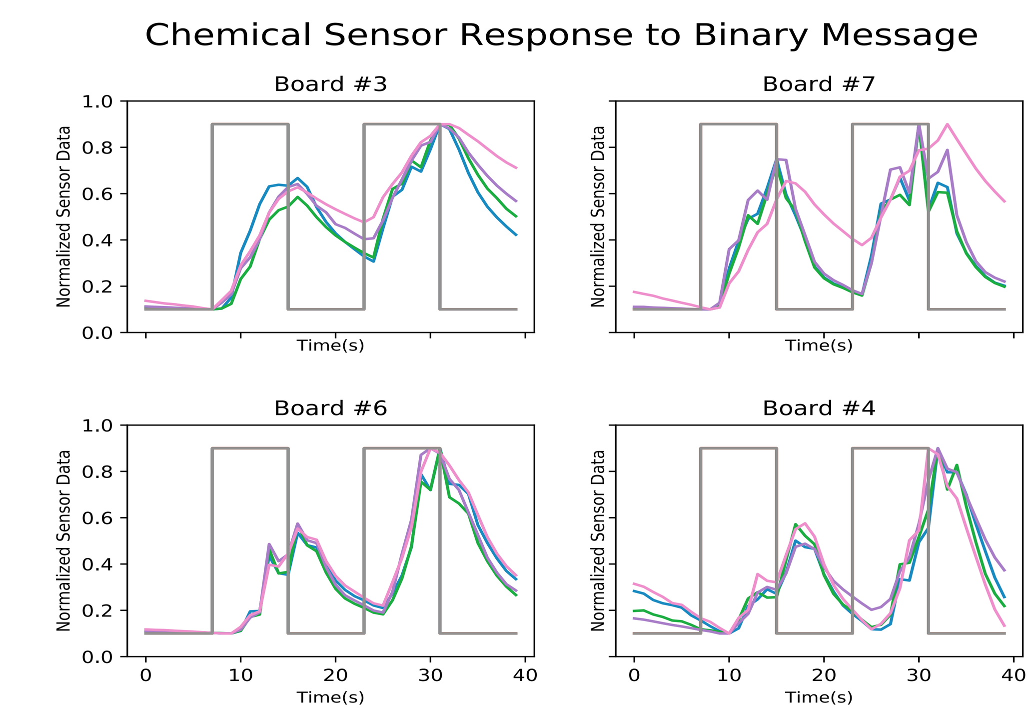
Software

All the code is written in Python using basic scientific packages
Using public libraries to communicate with the ADC and DAC
Using the Paramiko library for SSH and SFTP management of the Pis



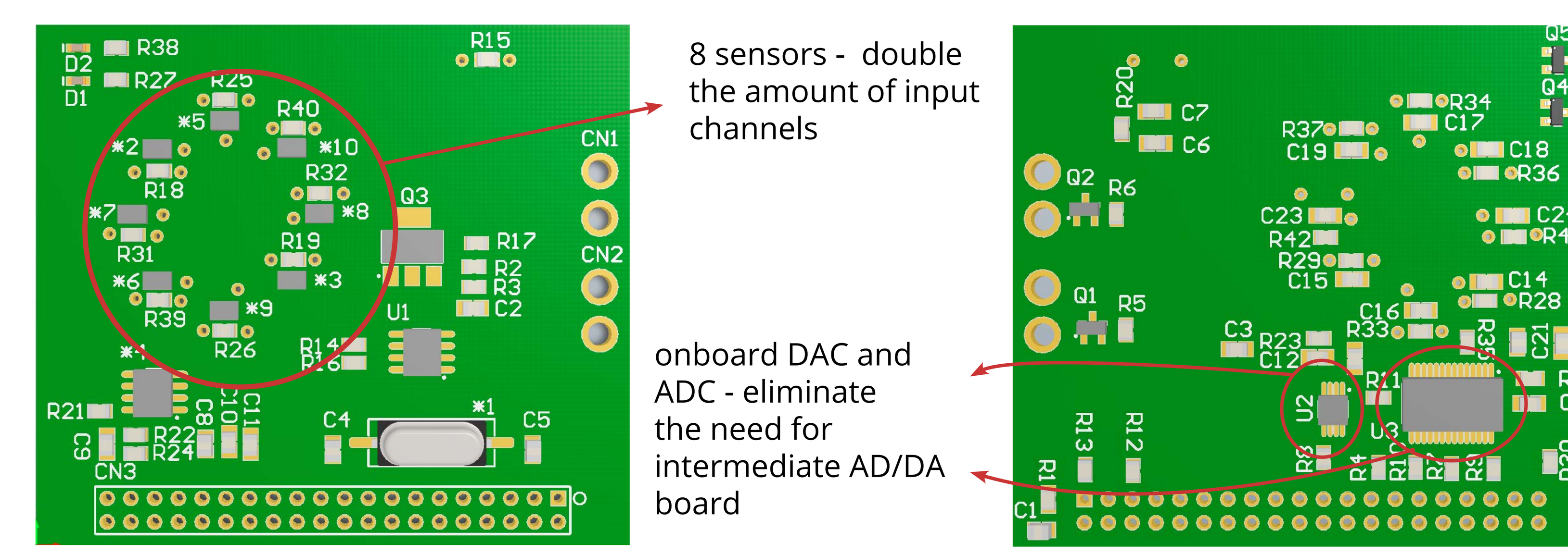
Experimental Setup

Results



Input message vs signal from sensors

	Before Software Change	After Software Change	After New Sensor Unit
Maximum Sample Rate	6.5 hz	8.5 Hz	8.5 Hz expected
Sensor Settling Time	0.6 s		
Sensors per Module	4		8
Simulataneous Modules	9	>50 theoretical	



New sensor unit PCB rendering

Ongoing Work

- Scaling up number of sensor units
- Assembling and testing new PCBs with 8 sensors and onboard DAC/ADC
- Collecting data in order to to train neural networks to recognize presence of patterns in time series

Acknowledgements

Much thanks to my advisor Jacob Rosenstein, as well as UTRA, DARPA, Christopher Rose, Pratistha Shakya, and Chris Arcadia.