

Dennis Moore

<https://github.com/dennis-moore>

Email : denmoore93@gmail.com

Mobile : 281-608-9655

EDUCATION

- **Texas A & M University**

Bachelor of Science in Electrical and Computer Engineering

College Station, TX

Aug. 2011 – Dec. 2015

EXPERIENCE

- **SOE**

Embedded Software Engineer

Houston, TX

May 2018 - Present

- Optimized large C code base to minimize code size, improve performance and remove warnings. Reduced code size by 25%, build time by a factor of 6 and removed 40,000+ warnings.
- Fixed C makefile to cross-compile code with the ARM GCC toolchain inside a Docker container.
- Developed Python scripts to integrate with Unix tools(nm, readelf, map file) and generate metrics on the code base. Added these tools to the Docker + Gitlab CICD environment to auto-generate metrics upon successful build.
- Improved performance of the eCompass by increasing the RTOS task rate and sensor update rate, and adding low-pass digital filters. Configured accelerometer and magnetometer sensors over I2C.
- Updated a unit test project to receive and log sensor data from an embedded device over the serial port. This data was analyzed and documented with Jupyter Notebooks, and used to design the low-pass filters for the eCompass.

- **Cognizant Technology Solutions**

Software Engineer

Dallas, TX

Jan 2016 - April 2018

- Worked in an Agile environment to design, develop, and test production software. Developed and maintained various Java, Python and C# applications that were used across many teams.
- Developed multiple Java applications that utilized REST APIs to improve upon and add features to our CICD environment.
- Maintained Windows virtual machines running Jenkins used for regression testing and CICD. Developed Java and Python code to automate parts of deploying new VMs. Provided VM support during production deployments.
- Created a couple of C# web apps hosted internally on a Windows VM with IIS. Both applications used .NET, Bootstrap, Javascript and MySQL.
- Constantly learned new technologies and worked on new ways to enhance the Selenium and TestNG testing framework through software innovations. Wrote a Java application to speed up regression execution, and improve testing coverage and efficiency.
- Gave presentations and lead training sessions on the applications I built, framework enhancements I made, and our current technology stack.
- Supervised team and lead regressions during my boss's absence. Interviewed candidates for open positions to assess their technical ability.

- **SOE**

Embedded Software Engineer Intern

Houston, TX

Summer 2015

- Defined a low level interface between two processors (TI C2000 and Arduino) to allow for SPI communication between the devices. Designed a level shifter to match their voltages(3.3V and 5V). Implemented error checking to throw away bad data packets.
- Researched and experimented with CAN bus communication between a Raspberry Pi and TI C2000.
- Wrote Python code to plot magnetometer data coming over serial port.

- **AMBER Robotics Lab**

Student Researcher

College Station, TX

May 2013 - July 2014

- I proposed a hardware system to test a nonlinear, autonomous cruise control system for cars that was being simulated in MATLAB. The idea was to port the control system to an Arduino, and use the Arduino to control the Servos of a gas powered RC car. This allowed the simulation and theory to be tested on hardware in the real world.
- Interfaced optical encoders with the Arduino over I2C and used these to compute wheel velocity. Applied a digital filter to this.
- Helped with the electrical design(microcontroller and sensors). Wrote a portion of the control system to feedback wheel velocity based on the encoder readings and drive the motors with PWM signals.

PROJECTS

- **NCOB:** *Spring Java web application for connecting and remotely controlling robots through an online interface*
 - Frontend created with HTML, CSS, jQuery and Thymeleaf. Backend created with Spring MVC and MongoDB. Deployed with Docker and Gradle.
 - C client code uses ZeroMQ to communicate with JeroMQ message broker on the web server. Google protocol buffers are used for message serialization.
- **AggiE-Challenge:** *Autonomous UAV that explored and mapped an unknown environment*
 - Setup Robot Operating System (ROS) environment on the Raspberry Pi. Developed TCP socket code in Python to communicate between the Pi and a host machine. Later added ROS to the host machine to ease wireless communication and add SLAM capabilities.
 - Interfaced ArduPilot flight control system with the high level control system on the Pi over Serial. Developed control system on the Pi to relay sensor data from the ArduPilot to the relevant ROS nodes, switch between flight modes (altitude control, etc.), and drive the quadcopter with user input.
 - Wrote device driver in C to interface Lidar sensor with the Pi. Later ported to ROS to ease integration with the rest of the system.
 - Developed Python code using OpenCV to visualize real time IMU data as a 3D cube to represent the attitude of the quadcopter. This was used to analyze the IMU and design a filter.
- **Senior Design:** *Microcontroller system for cars designed to monitor the environment and driver vitals*
 - Wrote serial communication drivers in Python and C to communicate between processors. Implemented memory mapped IPC to communicate between processes.
 - Sampled EKG sensor with ADC using timer interrupts and processed and filtered digital signal to determine heart rate information. Interfaced with LiDAR sensor over I2C to receive distance information for the control system. Utilized pin interrupts and Hall effect sensors to calculate wheel velocity.
 - Developed a control system based on kinematic equations and EKG/LiDAR/Hall sensor inputs to send commands and alerts over the serial link to the bluetooth interface.