Last Modified: 8/26/2019

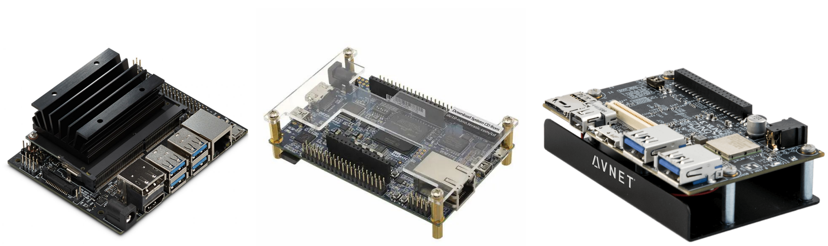
Project Member: Kowin Shi (CV), kss223@cornell.edu

Supervising Professor: Bruce Land, potentially Hadas Kress-Gazit/Joseph Skovira

Project Title:
Hardware Accelerated Mobile Robot Manipulator Platform with Real-World Applications

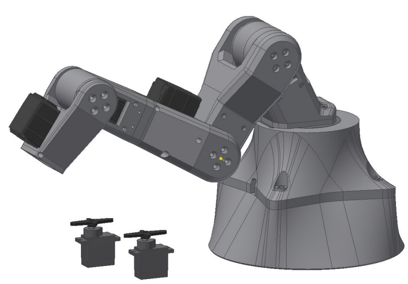
Project Fields: Electrical/Mechanical Hardware, Design Engineering, Hardware Acceleration (FPGA + CUDA), Robotics, Computer Vision, Machine Learning/Neural Nets, Controls Engineering

Project Description: The goal is to create a custom 4-wheel-drive mobile robot platform that is open-source, relatively affordable, compact in size, and have enough compute horsepower for high speed SLAM and path planning for both the mobile robot and the manipulator arm. The robot will have 2 quick release docks for attachments at either end, to fit the manipulator arm and a miniature cyclone-dust-separating vacuum module. It will rely on ROS for its software framework, and there will be plenty of custom code to interface with the various hardware systems onboard.



The robot will carry an Nvidia Jetson Nano for CUDA acceleration, an Intel/Altera DE10-Nano for FPGA acceleration, and a Xilinx Ultra96 (Zynq Ultrascale+) board for additional FPGA processing power. The two final high-level functionalities I have in mind currently are: 1. High speed robot vacuum with automatic emptying via manipulator arm and 2. Vision based pick and place objects with manipulator arm. A potential third high-level functionality could be Professor Skovira's UV cleaning project, depending on time.

Hardware: Almost everything will be designed and made from scratch, from the mobile robot to the 6-DOF manipulator arm to the vacuum module. The arm has already been designed by me and is currently in the manufacturing phase (3D printing):



Last Modified: 8/26/2019

The vacuum module is still in the research phase, and I am referencing the following projects: DIYson, Dual Stage Cyclone.

The mobile robot base is currently being designed, and is expected to be completed (manufacturing-wise) by the end of September. It will be driven by 4x NEMA17 stepper motors (for more accurate control inputs) with TMC2130 drivers (for quietest operations), and powered by 16x 18650 lithium-ion cells (4s4p) I salvaged from a Tesla battery module. The goal is to make it autonomously rechargeable, with it being able to return and mate with a wireless docking station when power is low.

In terms of the drivetrain, I intend on exploring both holonomic and nonholonomic drives as the 4-wheel-drive capability will allow for either. This will be very interesting to see in terms of the differences in control algorithms.

In terms of sensors, I intend on using the RPLIDAR A1M8 lidar for mapping and navigation, and either an intel realsense camera or just a raspberry pi camera for computer vision. Bump sensors may be added depending on complexity/necessity.

Software: The current plan is to use the DE10-Nano for mapping/navigation, as it is able to leverage its FPGA for massive parallelization of particle filter in the FastSLAM algorithm, and process the raw lidar data with as little delay as possible (lidar is max 8khz sampling rate). This results in a ultra-high update frequency for better localization performance, and thus allows for higher velocities in its trajectory. This requires Verilog/System Verilog designs on the FPGA, and C++ interface on the HPS side for the lidar interface. I am leaning towards using ROS to simplify the SLAM map storage/visualization functionalities, although finding an appropriate linux OS may require some research (since it has to work with the ARM SoC).

With the safety critical computation offloaded to the DE10-Nano, I am planning to use the Jetson Nano as the high-level planner for the whole robot to issue commands and dictate trajectories, as well as perform the less critical controls of the 6-DOF manipulator. The manipulator computation may get offloaded to the Ultra96 board, if it becomes too computationally taxing. The Jetson Nano will most certainly be running ROS, so some combination of python/c++. Some of the vision components may also be run on the Jetson, as it has 128 Maxwell GPU cores and Nvidia specifically advertised it for vision/robotics applications.

The Xilinx board will be used for further hardware acceleration, as it is has one of Xilinx's newest Zynq FPGA processors combined with a quad-core HPS. In particular, it will be used for neural net acceleration in object detection algorithms. The FPGA is large enough to also run parallelizable path planning algorithms for the manipulator arm, if deemed necessary.

In terms of inter-computer networking, both the DE10-Nano and the Jetson Nano have ethernet ports, while the ultra96 supports ethernet through usb. This part is still TBD upon more discussions with Professor Land, but most likely all 3 will communicate with UDP protocol through a network switch, and either the ultra96 (which has wifi built into the pcb) or the Jetson Nano (has wifi M.2 expander card) will handle external communications such as to another computer for visualization and GUI.

Specific algorithms that I am looking to implement are: FastSLAM, RRT/Visibility Roadmaps, Bug, Nonlinear MPC, Forward/Inverse Kinematics, RCNN

Preliminary Sketch: