# Ashar Alam

ashar1@stanford.edu linkedin.com/in/alam-ashar +1-650-283-4075
https://github.com/asharalam11 https://asharalam11.github.io

## EDUCATION

### STANFORD UNIVERSITY
MS in Mechanical Engineering
Mechatronics, Robotics & Controls
June 2020 | Stanford, CA
Cum. GPA: 3.83

### KAIST
BS in Mechanical Engineering
August 2018 | Daejeon, South Korea
Magna Cum Laude
Cum. GPA: 3.91

### GIIS QUEENSTOWN
May 2014 | Queenstown, Singapore

## COURSEWORK

- **(ME218)** Smart Product Design
- **(AA274)** Principles of Robotic Autonomy
- **(ME227)** Vehicle Dynamics and Controls
- **(CS 223A)** Introduction to Robotics
- **(CS107)** Computer Organizations & Systems
- **(CS106B)** Programming Abstractions in C++
- **(EE263)** Introduction to Linear Dynamical Systems
- **(ENGR205)** Control Design Techniques
- **(CS238)** Decision Making Under Uncertainty
- **(EE364)*** Convex Optimization

**(Courses to be done in Winter' 2019)*

## SKILLS

### PROGRAMMING
• C • C++ • Python • HTML/CSS
• ROS • MATLAB/Simulink • Docker
• Assembly • Unix Tools and Scripting
• CARLA • Unreal Engine • Blender • Git

### ELECTRICAL
• CAD - Altium Designer
• Equipment - Oscilloscope, Saleae Logic Analyzer, DMM, Function Generator, Soldering

### MECHANICAL
• CAD - SOLIDWORKS, AutoCAD
• CAE - ANSYS, OpenFOAM
• Manufacturing/Prototyping - Conventional Workshop machines, Laser cutting, FDM/SLA 3D Printing, Clean Room, Photolithography

## PROFESSIONAL AND TECHNICAL EXPERIENCE

### COURSE ASSISTANT | Stanford University
Winter 2020
- **(AA274B/ CS237B)** Principles of Robotic Autonomy II
- **(ME210)** Introduction to Mechatronics

### LUCID MOTORS | Autonomous Driving Software Intern
Jun 2019 - Sep 2019 | Newark, CA
- Tested control algorithms on multiple driving simulation platforms using C++ and Python for validating controller design for ADAS and AD scenarios
- Developed middleware interface between simulation softwares and controller architecture and implemented controllers for ADAS systems
- Customized simulation software environment for validating and testing vehicle dynamics and implemented control algorithms for various scenarios encountered while driving autonomously on highways

### SMART PRODUCT DESIGN | Mechatronics - ME218
Sep 2018 - Jun 2019 | Stanford University | Prof. Ed Carryer
- Built intelligent electro-mechanical systems and embedded hardware and software with experience in:
  - Efficient event-driven software design using state machines
  - Use of SPI, I2C and UART for inter-process communication
  - Circuit design for range detectors, photodiodes/photoresistors, accelerometers, motor drivers, voltage regulators, DC, Stepper Servo motors
  - Use of equipment such as oscilloscopes, logic analyzers, DMMs etc.
- Built an arcade game, an autonomous ball collecting robot and a remote controlled hovercraft (using XBee for RC communication) with event-driven code in C on a TI (ARM 32-bit) microcontroller and assembly code on a PIC

### AIR TRAFFIC CONTROL - REINFORCEMENT LEARNING | CS238
Sep 2019 - Dec 2019 | Stanford University | Prof. Mykel Kochenderfer
- Developed a Pygame environment for simulating aircrafts using Python3
- Modelled the air craft collision problem as an MDP (Markov Decision Process) and implemented Reinforcement Learning algorithms like Q-Learning and SARSA Learning to generate optimal policy for our aircraft agent.

### AUTONOMOUS DRIVING - SIMULATION & EXPERIMENT | ME227
Apr 2019 - Jun 2019 | Stanford University | Prof. Chris Gerdes
- Generated a velocity and acceleration profile, and calculated lateral forces based on vehicle modelling to traverse an oval path in a given time
- Designed and calculated gains for a Lookahead controller and LQR controller with feedback to adhere to generated profiles
- Simulated controller performances in MATLAB and implemented the aforementioned controllers on a Golf GTI which drove itself autonomously around its designated oval path in a parking lot

### AUTONOMOUS DRIVING WITH A TURTLEBOT | AA274
Jan 2019 - Mar 2019 | Stanford University | Prof. Marco Pavone
- Developed a ROS package for a TurtleBot3 Burger to autonomously explore a mock environment where it needed to "pick up and deliver" food items by implementing algorithms in Python to develop an autonomy stack including modules for perception, localization, motion planning and controls
- Used a costmap overlay and a modified Djikstra algorithm over a map generated by EKF-SLAM package to find the closest unexplored part of the map