Team 11 - Project Robofuel

A Better Alternative to Traditional Fueling Infrastructure

Authors:
Steven Hellem

Ahmed Abdel-Gwad

Michael Gomez

Jordan Moore

Supervisor:
Professor Neal Levine

Date Submitted: May 3, 2021

CONTENTS

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

An automated robotic vehicle fueling system is proposed to increase accessibility, safety, and eliminate fomite Coronavirus transmission.

There are numerous problems associated with how traditional gas stations are operated in modern times. First, gas pumps are operated manually and as such require many people to touch the same handles and keypads. This makes these physical elements potential germ and disease vectors. Another corollary sub-problem is associated with spillage which can occur whenever someone ”tops-off” their gas tank. Various hydrocarbons containing harmful chemicals such as benzene, a known carcinogen that is detrimental to human and environmental health, escape into the atmosphere and ground soil. A security concern regarding gas stations is credit card fraud. Personal information such as account and user identification is embedded in the magnetic strip behind credit cards and is stolen by criminals using card skimmers. There are millions of dollars lost every year by gas stations due to this issue. With the quick advances in battery technology, electric and autonomous vehicles are on the rise. This poses an infrastructure problem as typical gas stations are slow to adapt to the shift toward electric vehicles. Because of this, recharge stations are being placed in parking lots at retail stores, schools etc. A fully automated gas pump would solve all of these problems as it reduces viral transmission, eliminates credit card skimming, cancels human error of operation and increases accessibility to the electric car market.

The system will capitalize on a System-On-Chip architecture, which includes an FPGA and computer processor on the same chip. This means that the design will take advantage of the parallel nature of the FPGA to rapidly process image data, high speed data transmission between FPGA and CPU, and the flexibility of an embedded software system for controls and further image processing.

The physical structure and kinematics will be capable of recreating the range of motion of a gas station attendant or self-service user. This means that the structure must be capable of changing position in the xyz plane, as well as adjusting the angle of the nozzle to facilitate insertion. The user will park in a predetermined area within the range of the motion of the kinematics system.

The movement of the fueling arm must not cause damage to a vehicle. Proportional velocity and position control lends the risk of overshooting when moving to a specified position. Overshooting the movement of the fueling arm risks scratching or denting a vehicle while moving to one of the positions specified for fueling. Proportional integral derivative (PID) control will provide stability of movement by adjusting the signal response to position feedback and lowering the chance of a large response the closer the arm is to its target position. This feature can be measured by recording the proportional, integral, and derivative error measured while in use and the change in response as the arm reaches a target position or velocity.

The computer vision is the heart of the design and the system. The final version of the system will be capable of streaming live video data from a camera attached to the robotic arm to the embedded system, and informing the controls system to make decisions regarding robotic arm movement in real-time. This final version will be developed in an iterative fashion. First, a software-only system will be designed, capitalizing on the Intel Realsense SDK 2.0, and associated OpenCV wrappers. This software system will handle all the image processing, then pass data to the FPGA to inform the motor control systems. After successfully implementing a software solution, a hardware-based solution for image processing will be implemented to improve performance further.

A mobile application will be implemented to handle payment, display pump usage information and waiting times, provide directions to the nearest station, and manage user accounts. This is a full-stack endeavor, that will require a mobile user interface, multiple databases, and a stable API available for developers to access. The application is expected to provide a seamless, aesthetically pleasing, and high-performance experience from an empty tank to full tank.

Abstract—**In light of a global pandemic, effecting more than 20 million people and lasting longer than 6 months, there is an ever growing need to solve how traditional gas pumps are operated today. There are many sub-issues along with slowing the spread of COVID-19. Some of these are limited accessibility, identity fraud, human operating error and harmful agents absorbed by humans and introduced into the atmosphere. The current infrastructure of gas stations is catered toward gas operating vehicles. An automated gas pump utilizing a System-On-Chip architecture, containing both an FPGA and computer processor is proposed to solve the issues associated with traditional self-service gas pumps. The design will utilize the parallel components of the FPGA allowing for rapid data transmission, fast image processing and increased flexibility. The design will have 4 degrees of freedom comprised of 3 linear axis and 1 rotational axis. It will be constrained by, 1000 millimeters of x axis movement (horizontal movement), 600 millimeters of y axis movement (vertical adjustment), and 1000 millimeters of z axis movement (reach). The end effector axis will require a minimum of 90 degrees rotational motion. The design will be integrated with a phone application for convenient payments, security, accessibility and a hands-off experience, while utilizing NFC technology. Along with automation, the design will include mechanical on/off switches in case of mechanical or electrical failure. Proportional integral derivative (PID) control will be implemented to provide stability of movement by adjusting the signal response to position feedback thus lowering the chance of a large response upon the arm's approach to the gas hole.**

Keywords— SARS-Cov2, Coronavirus, COVID-19, automated, gas-pump, top-off, spillage, credit-card fraud, Robofuel, refueling

I. INTRODUCTION

According to the US National Institute of Health, COVID-19 can live up to three days on stainless steel and plastic surfaces[5]. This means that frequently used vehicle fuel pumps are a constant vector for the virus transmission. A contact-less, fully robotic fuel pump is proposed, so as to limit the transmission of contagious diseases through surface contamination, reduce environmental contamination due to fuel spillage, eliminate credit card fraud in the form of fuel pump ”skimming”, and increase accessibility for disabled customers. The customer will drive alongside the pump, pay via the mobile app, and the robotic pump will open the gas tank, then insert a nozzle and fuel the vehicle.

Existing research shows that in the event of a pandemic, attitudes towards public health measures such as quarantine differ greatly through varying demographics ranging from fear, to denial, to compliance. Even in the case of compliance in a quarantine, rigor of isolation is applied unevenly internally throughout different households. Skepticism towards public health institutions and media reports contribute to an unwillingness to comply. Transitioning fuel pumps from manual operation to full automation eliminates the chance of disease transmission through surface contamination and aerosol transmission, even if sections of the populace are unwilling to comply with mandated public health measures.

The American Disabilities Act (ADA) expressly states that self-serve gas stations provide equal access to their customers with disabilities. However, a service station or convenience store is not required to provide such service at any time that it is operating on a remote control basis with a single employee, but is encouraged to do so, if feasible. This means that a disabled customer may very well encounter a situation where they could not access fuel. The proposed system eliminates this possibility.

The Federal Trade Commission (FTC) announced over 124,000 reports of payment card fraud in 2018, with combined losses of approximately 287 million dollars. From a security standpoint, a full-service fuel station is superior because at no point should the customer be tampering with the pump or even outside of their vehicle. An app-based payment system would negate usual security concerns with physically swiping a credit card, and security concerns will lie purely in the realm of software and encrypted communications, which require a much greater degree of technological sophistication than hardware skimmers.

Environmental contamination due to fuel spillage and groundwater seepage is a major concern as it not only has negative effects on the earth but is also a public health hazard. Research indicates that small spills caused from user error and design flaws have a negative impact on the environment. There is a portion of the spills that penetrate the concrete slabs surrounding gas pumps and are absorbed into the soil and groundwater. Spillage can also be transported via foot traffic and rain sometimes into natural occurring water bodies as well as adjacent soil. Whenever an individual fills their gas tank they are introducing hydrocarbon vapor into the atmosphere due to a phase change from liquid particles into the gaseous state. This hydrocarbon vapor contains known carcinogens and toxins which are detrimental to human health as well as ground level ozone. An automated gas pump system would mitigate most user errors and the amount of contaminates introduced into the environment would be accurately quantifiable and greatly reduced.

The exposure to benzene molecules found in gasoline, also pose a risk of absorption to gas station workers. Benzene, as it stands, is considered in the top 10 chemicals of greatest toxicological importance. Likewise exposure to this compound is a public health issue. The primary way that benzene is ingested by humans is by respiratory means. Dependent to the level of exposure there are symptoms that correlate to benzene poisoning. Some of these are; dizziness, headache,loss of consciousness and death. It can be seen in blood counts that even low amounts preceding 1 ppm is unhealthy. For gas station attendants this poses an increased risk as they often work 8 hour days and 6-7 days a week. Their tasks also play a part in their level of exposure. Some of these include checking and fuel pumps, testing fuel samples received, measuring fuel levels in underground tanks, and unloading the fuel from the delivery trucks. There is a study that was published July 23, 2020 by Int J Environ Res Public Health that evaluates the observes the exposure to different groups of gas station workers as well as reported symptoms in correlation to their specific tasks. It also accounts for the location of the gas stations to better calculate correlation and causation. An overall solution to most of the problems with gas pumps may be to implement full-service with gas station attendants. An alternative full-service solution using robotics can address the risk to the gas station attendant in this scenario.

To address the myriad problems associated with self-service gas stations, a design basjed on computer vision and a robotic arm with 4 degrees of freedom will be implemented to facilitate an automated gas pump. The system will take advantage of an Intel System-On-Chip architecture, which includes an FPGA and computer processor on the same chip. This means that the design will take advantage of the parallel nature of the FPGA, rapidly processing image data, the high speed data transmission between FPGA and CPU, and the flexibility of a software system for controls and further processing.

The vehicle refueling industry has become vastly more complex compared to its original design. Knowledge of environmental processes, vehicle types, fuel types, and user scenarios have all changed since the core design of a fueling station emerged. The redesign and upgrade of the typical fueling system can achieve significant improvement by integrating technologies widely available today. An automated refueling system will allow a deterministic improvement that can provide accommodation to stakeholders not currently addressed in refueling systems.

There are many stakeholders to address when considering the problems caused by conventional gas stations. Some of the critical stakeholders to consider would be drivers, fuel station owners, environmental stakeholders, and the population in general. There are financial, environmental, accessibility, and health risks that affect this multitude of stakeholders. An automated fuel station design hopes to address these risks affecting the stakeholders to the refueling industry.

The design proposed aims to reduce the issues involved with human interaction in the fueling process in a cost effective and robust way. There are some companies such as Fuelmatics or Electify America that have attempted similar solutions to an automated fueling system. These solutions present high barrier to entry costs in the order of ten to fifty thousand dollars conservatively. The barrier to entry with a fuel pump as expensive as this makes widespread adoption a non starter for many companies in the fossil fuel industry.

The lower barrier to entry is driven through the use of hardware accelerated solutions where possible and reliable methods of actuation. Stereo vision camera technology is able to provide a high level of relative position feedback in a simple form factor. Due to the highly variable nature of vehicle positioning, relative positioning will be crucial to the design of such a system. Current technology such as the Intel Real Sense technology offloads the large amount of processing needed for this type of depth mapping. This form of hardware accelerated solution lowers the cost in the embedded system needed for controlling an automated fuel system. Using an FPGA in combination with a hard processor system will also allow for lowering the costs associated with computing power. Reliable lead screw actuation will lower maintenance cost, make local positioning more reliable and highly repeatable.

Choosing the right combination of cost effective and newly available technologies will allow the ability to implement an autonomous fueling solution for the twenty first century. The automated fuel station will include stakeholders such as disabled drivers, those who are vulnerable to health issues such as the Coronavirus, the environment, and financial stakeholders in the design.

II. DISCUSSION OF SOCIETAL PROBLEM

A. Damage Done by Escaped Fuel Droplets

In the last 6 years there have been studies focusing on smaller scale issues surrounding escaped droplets from gas pumps. This at first was considered too small scale to be studied in depth. However, research indicates that small spills may actually have a negative impact on the environment. The researchers' estimate that around 1500 liters of gasoline fall onto the concrete slabs next to gas pumps every 10 years. There is a percentage of that volume that penetrates the surface of concrete and is absorbed into the soil as well as groundwater. This is problematic for those using wells for their water supply as gasoline contains harmful chemicals such as benzene which is a carcinogen. Two professors at the Department of Environmental Health Sciences have created a mathematical model which demonstrates the amount of gasoline that seeps into the concrete as well as the amount that vaporizes into the atmosphere. Contrary to popular belief if no stain is left behind from spillage, some droplets do in fact seep into the concrete and into the ground soil. It was found with this model that a statistically significant amount of gasoline droplets does penetrate the concrete surface and can have a long-term effect. There are still holes in the research of health effects relating to living near gas stations. However, this could likely become a problem due to the fact gas stations are moving toward higher volume distributors. The leakage is even more troublesome when introducing rain into the equation. For if a gasoline spill mixes with rain it can then travel into adjacent soil or be introduced into natural occurring water bodies. Foot traffic can also help these contaminants travel to more porous locations. There are currently no regulations surrounding this phenomenon even though studies have shown non-negligible effects [7].

Any time there is a transfer from liquid hydrocarbons from one medium to another there is a risk for environmental contamination. For example, when an individual fills their gas tank they are introducing hydrocarbon vapor into the atmosphere due to inter-phase mass-transfer from the respective liquid into its gaseous state. Likewise, when there are spills of fuel the droplets are fall on the surrounding ground and this can be observed from stains on concrete at fuel stations. This appears to not be a rare event according to studies performed by Mueller, Wixtrom and Brown [7]. The average liquid gasoline released on average when refueling a vehicle is 0.9g and 2.4g. Once being introduced to the ground liquid hydrocarbon can pierce the grounds membrane and be introduced into the soil and ground water.

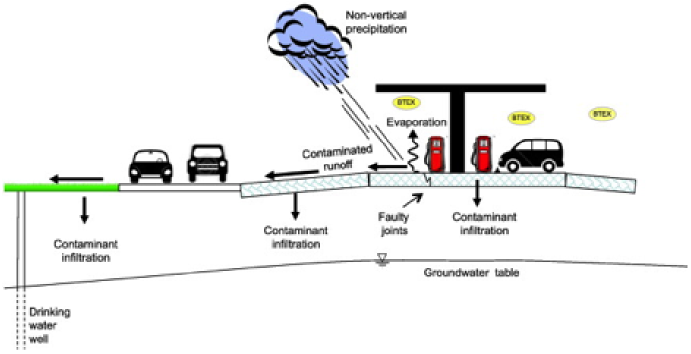


Fig. 1: Conceptual model of fate and transport of spilled gasoline

There are multiple ways in which this spilled hydrocarbon may affect humans and the environment. The first is that the vapors from hydrocarbon can be inhaled. The second is that liquid hydrocarbon can be transported via runoff water can be ingested or introduced into eye cavities as more common with children playing near sites. This can also occur when the diluted water fuel mixture penetrates into the soil. Another example is when fuel makes it to the ground water supply and is ingested. This is especially the case in more rural areas of living as well systems or more relied on. These hydrocarbon contaminates are harmful as they contain compounds such as benzene, toluene, ethylbenzene and xylenes which are known carcinogens and toxic. These compounds are not only detrimental to human health but also to the ground-level ozone. This is why Stage I and II vapor recovery systems have been added onto gas station dispensers to mitigate some of these effects however the vapor pressure of fuel is only regulated during the Summer. There are no extensive studies or guidelines as of now for the purpose of concrete pads and what amount penetrates the concrete slabs surrounding fuel sites. This is why Markus Hilpert and Patrick Breysse have developed a mathematical model to quantify the amount of fuel infiltration into the concrete and evaporation into the atmosphere in order to explain disappearance of spilled gasoline droplets from the ground. It is also useful in modeling pathways for human exposure and estimated lifetimes of such droplets.

The basis for the mathematical model is derived from the mass balance equation because it correlates the rate of change of the total liquid mass to the potential rate of evaporation.

$$\frac{dm_l}{dt} = -\pi R_d^2 E_{po} \tag{1}$$

Where ml, to the potential rate of evaporation, Epo 0 [kg m 2 s 1] and Rd is the drawing radius, and t is time. There is two components to the total liquid mass:

$$m_l(t) = m_{sd}(t) + m_{in}(t) \tag{2}$$

where msd is the mass of the sessile droplet sitting on the external surface of the porous medium, and min is the mass of liquid that infiltrated into pores.

An automated and touch-less gas pump system would eliminate user error caused by individuals ”topping off” their gas tanks causing unnecessary droplets to accumulate on the concrete slabs surrounding the pumps by allowing designated sensors to determine when the tank is full and then stopping flow of fuel to the injector. This will mitigate the risks of ingestion, residual run-off into ground water supplies, and to the environment. [7]

B. *Benzene Absorption by Gas Station Workers*

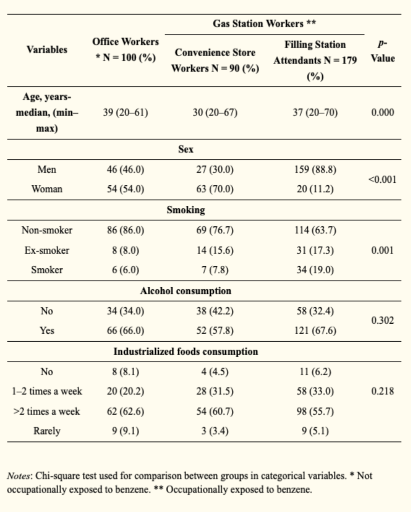
A study done in two urban locations in Metropolitan Rio de Janeiro, performed by Int J Environ Res Public Health, shows the exposure to Benzene and reported symptoms. The two geographic locations used for this study showed no differences in the specific tasks done by workers. The differences that were present were due to the gas station locations. The greater urbanized areas with large buildings and more vehicle traffic exhibited less air dispersion which explains some of the higher t,t-MA urinary levels in workers. The control group in this experiment was comprised of office workers in administrative positions as they are not exposed to the fuel nor the Benzene. Only workers at the gas stations who have worked their for more than 6 months were included in the study [10].

From these workers, a questionnaire was administered including complaints, symptoms and signs controlling for preexisting conditions. After all 8 hour shifts, urine samples were collected from the workers. Due to possible dilution of urine, samples with Creatinine levels of ¡0.3 g/L and 3.0 g/L¡ were excluded. With the valid samples, a multivariate analysis was performed for Benzene exposure. The variables controlled for are shown below in Figure 4. The distribution and comparison of t,t-MA urinary levels (mg/g Creatinine) in office workers and gas station workers exposed to Benzene is shown below in Figure 5. The symptoms associated with Benzene poisoning are shown below in Figure 6 [10].

This study demonstrates there are lower levels of Benzene in the office workers and higher rates in gas station workers who also reported adverse symptoms. An automated gas pump would completely mitigate unnecessary exposure to the harmful compound Benzene [10].

C. *Payment fraud mitigation*

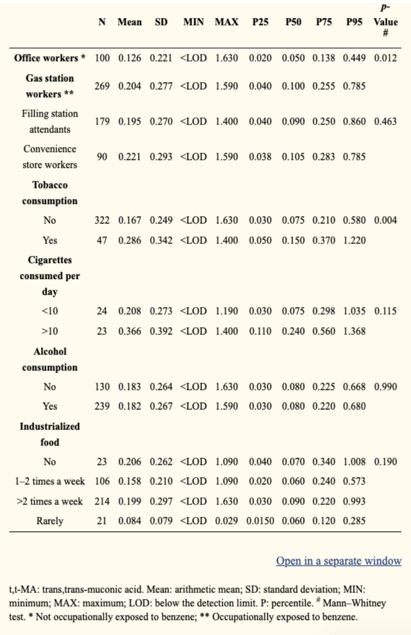
Gas pump payment fraud results in millions of dollars in losses per year in total despite costing stores an average of $700 per year. When payment card information is captured and compromised, it can be used to commit multiple types of fraud, all of which are types of identity theft. These include fraudulent online payments, ”cloning” the information onto the magnetic stripe of a blank credit or debit card, or selling the information on the black market to criminal actors. [2] Crucial user identification and account information is encoded on the magnetic stripe of payment cards. Criminals use ”skimmers” to to capture data from payment cards as cards are swiped. Gas stations are frequent targets for this kind of criminal activity, as they are minimally supervised late at night, and a criminal can extremely quickly and discreetly install one of these devices.



Variables	Office Workers * N = 100 (%)	Gas Station Workers **		p-Value
		Convenience Store Workers N = 90 (%)	Filling Station Attendants N = 179 (%)	
Age, years-median, (min–max)	39 (20–61)	30 (20–67)	37 (20–70)	0.000
Sex				
Men	46 (46.0)	27 (30.0)	159 (88.8)	<0.001
Woman	54 (54.0)	63 (70.0)	20 (11.2)	
Smoking				
Non-smoker	86 (86.0)	69 (76.7)	114 (63.7)	0.001
Ex-smoker	8 (8.0)	14 (15.6)	31 (17.3)	
Smoker	6 (6.0)	7 (7.8)	34 (19.0)	
Alcohol consumption				
No	34 (34.0)	38 (42.2)	58 (32.4)	0.302
Yes	66 (66.0)	52 (57.8)	121 (67.6)	
Industrialized foods consumption				
No	8 (8.1)	4 (4.5)	11 (6.2)	0.218
1–2 times a week	20 (20.2)	28 (31.5)	58 (33.0)	
>2 times a week	62 (62.6)	54 (60.7)	98 (55.7)	
Rarely	9 (9.1)	3 (3.4)	9 (5.1)	

Notes: Chi-square test used for comparison between groups in categorical variables. * Not occupationally exposed to benzene. ** Occupationally exposed to benzene.

Fig. 2: Socio-demographic characteristics of office workers and gas station workers in Brazil



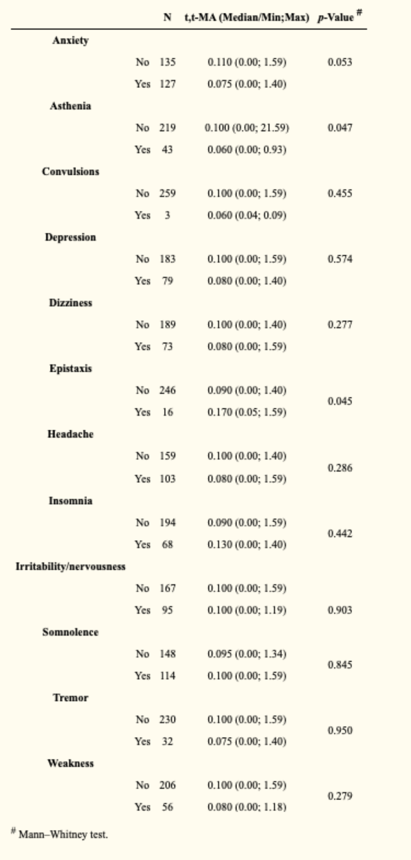
	N	Mean	SD	MIN	MAX	P25	P50	P75	P95	p-Value #
Office workers *	100	0.126	0.221	<LOD	1.630	0.020	0.050	0.138	0.449	0.012
Gas station workers **	269	0.204	0.277	<LOD	1.590	0.040	0.100	0.255	0.785	
Filling station attendants	179	0.195	0.270	<LOD	1.400	0.040	0.090	0.250	0.860	0.463
Convenience store workers	90	0.221	0.293	<LOD	1.590	0.038	0.105	0.283	0.785	
Tobacco consumption										
No	322	0.167	0.249	<LOD	1.630	0.030	0.075	0.210	0.580	0.004
Yes	47	0.286	0.342	<LOD	1.400	0.050	0.150	0.370	1.220	
Cigarettes consumed per day										
<10	24	0.208	0.273	<LOD	1.190	0.030	0.075	0.298	1.035	0.115
>10	23	0.366	0.392	<LOD	1.400	0.110	0.240	0.560	1.368	
Alcohol consumption										
No	130	0.183	0.264	<LOD	1.630	0.030	0.080	0.225	0.668	0.990
Yes	239	0.182	0.267	<LOD	1.590	0.030	0.080	0.220	0.680	
Industrialized food										
No	23	0.206	0.262	<LOD	1.090	0.040	0.070	0.340	1.008	0.190
1–2 times a week	106	0.158	0.210	<LOD	1.090	0.020	0.060	0.240	0.573	
>2 times a week	214	0.199	0.297	<LOD	1.630	0.030	0.090	0.220	0.993	
Rarely	21	0.084	0.079	<LOD	0.029	0.0150	0.060	0.120	0.285	

Open in a separate window

t,t-MA: trans,trans-muconic acid. Mean: arithmetic mean; SD: standard deviation; MIN: minimum; MAX: maximum; LOD: below the detection limit. P: percentile. # Mann–Whitney test. * Not occupationally exposed to benzene; ** Occupationally exposed to benzene.

Fig. 3: t,t-MA urinary comparison of office and gas station workers in Brazil

These skimmers come in a variety of form-factors. Refer to figures 5 - 9 [3] on page 6 of this report.



		N	t,t-MA (Median/Min;Max)	p-Value #
Anxiety				
	No	135	0.110 (0.00; 1.59)	0.053
	Yes	127	0.075 (0.00; 1.40)	
Asthenia				
	No	219	0.100 (0.00; 21.59)	0.047
	Yes	43	0.060 (0.00; 0.93)	
Convulsions				
	No	259	0.100 (0.00; 1.59)	0.455
	Yes	3	0.060 (0.04; 0.09)	
Depression				
	No	183	0.100 (0.00; 1.59)	0.574
	Yes	79	0.080 (0.00; 1.40)	
Dizziness				
	No	189	0.100 (0.00; 1.40)	0.277
	Yes	73	0.080 (0.00; 1.59)	
Epistaxis				
	No	246	0.090 (0.00; 1.40)	0.045
	Yes	16	0.170 (0.05; 1.59)	
Headache				
	No	159	0.100 (0.00; 1.40)	0.286
	Yes	103	0.080 (0.00; 1.59)	
Insomnia				
	No	194	0.090 (0.00; 1.59)	0.442
	Yes	68	0.130 (0.00; 1.40)	
Irritability/nervousness				
	No	167	0.100 (0.00; 1.59)	
	Yes	95	0.100 (0.00; 1.19)	0.903
Somnolence				
	No	148	0.095 (0.00; 1.34)	0.845
	Yes	114	0.100 (0.00; 1.59)	
Tremor				
	No	230	0.100 (0.00; 1.59)	0.950
	Yes	32	0.075 (0.00; 1.40)	
Weakness				
	No	206	0.100 (0.00; 1.59)	0.279
	Yes	56	0.080 (0.00; 1.18)	

Mann–Whitney test.

Fig. 4: Correlation between Benzene poisoning and symptoms

A mobile payment application that would interface with an automated gas pump would completely eliminate any incidence of fraud conducted by skimming, whether facilitated by intercepting PINs, magnetic stripe data, or installing 'man-in-the-middle' hardware to intercept essential data. This mobile application could be expanded to show pumps in use, malfunctioning or out-of-order pumps, usage statistics, wait times, and other pertinent data. As a back-up, the gas pump could implement NFC payment, either through NFC enabled payment cards or mobile wallets. Though these types of payment systems can be vulnerable to adversarial attackers, they are orders of magnitude more secure than physical magnetic card readers and PIN pads.

D. *Compliance with public health measures*

Historically, infectious diseases cause the human greatest death tolls. For example, the bubonic plague killed approximately 25% of the European population. [Scott, S. Duncan, C.J. Biology of Plagues: Evidence from Historical Populations. (Cambridge University Press, 2001).] Consideration to how human's respond to perceived threats and the downstream consequences of those responses is given in the development of this technology that is specifically aimed at mitigating the pandemic.

Fear is a central human response to perceived threats. Results of a psychological meta-analysis show that strong fear appeals produce the greatest behaviour change only when people feel a sense of efficacy, whereas strong fear appeals with low-efficacy messages produce the greatest levels of defensive responses. On the other end of the spectrum, people are susceptible to suffering from "optimism bias", where they underestimate their likelihood of contracting the virus and more likely to ignore public health measures. Public health communication measures must strike a balance between break-

Fig. 5: Front of a typical overlay skimmer

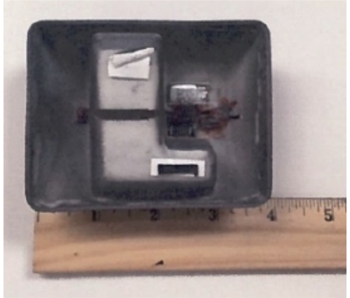


Fig. 6: Rear

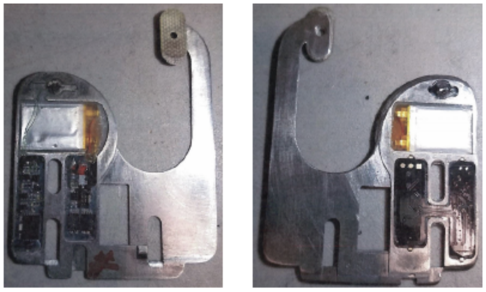


Fig. 7: Front and back of deep insert skimmer

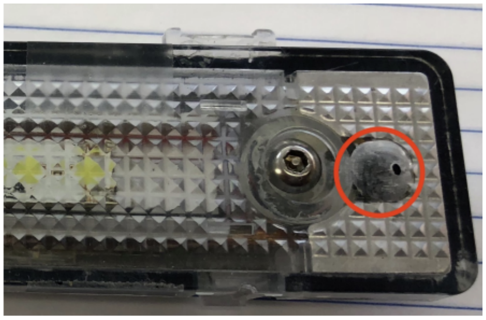


Fig. 8: Pinhole camera, used to record PIN entry

Fig. 9: Pin pad overlay used to collect PIN numbers

ing through optimism bias and avoiding feelings of anxiety and dread.

Humans are emotional creatures; research shows that emotion often drives risk perception, more so than factual data. [Slovic, P., Finucane, M. L., Peters, E. MacGregor, D. G. Risk as analysis and risk as feelings: some thoughts about affect, reason, risk, and rationality. Risk Anal. 24, 311–322 (2004).] In the case of strong emotional reactions, people are more likely to ignore probabilities and a problem's scope.

Slowing viral transmission during a pandemic requires significant shifting of behavior. Social norms, inequality, culture, and political polarization are all components of the social context that drives willingness to shift behavior in the face of a threat like the ongoing pandemic. People are heavily influenced by social norms - how they perceive what others are doing, and what they think others approve and disapprove of. Some research suggests that a larger proportion of behavior interventions can come not from direct effects on people who receive the intervention, but from indirect effects on their social contacts who copied the behavior. This is applicable to the design idea. If most people are staying in their cars while refueling, others will copy their behavior out of a desire to adhere to the social norm.

Perception of self as independent versus interdependent is also a function of culture. Western European and North American cultures that promote individualism are seen as independent, where other cultures that value family, tribe, or nation are seen as interdependent. The priority given to duty and obligations in Asian societies may convince individuals to adhere to social norms while suppressing personal desires. In contrast, Western European and North American cultures value the expressivity of self (kissing, hugging, direct argumentation), increasing likelihood of interpersonal transmission of the virus.

Political polarization among citizens has enormous consequences - decreased trust in the government and believing false information that can undermine social and economic relationships and impair public health. Partisans may receive different news because individuals can self-select polarized news sources or partisan 'echo chambers' or can communicate in ways that are associated with less cross-partisan information sharing. Partisanship and polarization is at an all-time high in America, and are major reasons for the nation's tepid response to the virus.

Psychological and sociological research [4],[5] shows that humans are not necessarily willing to comply with common-sense health and safety measures during an emergency pandemic such as wearing a mask and maintaining an appropriate distance from others, even when science supports the efficacy of these measures in preventing spread. This fact is particularly amplified in the time of COVID-19, where wearing a mask is now a fiercely-contested political statement in the United States.

Automating the gas pump completely takes the human option out of the equation - the customer is expected to remain in their vehicle while fueling. There will be no surface contamination because the entire process is automated and without contact. The chance of aerosol contamination is greatly reduced as customers are expected to stay in their vehicles.

E. Gas pumps as a disease vector

Infectious diseases often propagate though a multitude of vectors. An early study at the start of the Covid-19 pandemic aimed to demonstrate the half lives shown in Figures 10 and 11 of the virus with respect to surface transmission [5]. Stainless steel and plastic were among the common materials tested which are also the primary materials the average gas pump consists of. Infection from Covid-19 (among other infectious diseases) could potentially be lowered by reducing the amount of high traffic public surfaces. Health policies in response to the recent pandemic have had a focus to lower the amount of common contact surfaces as a means to curb viral transmission. Removing the need to handle common public surfaces such as a gas pump would be a logical extension to these health efforts.

An experiment was done in a controlled lab environment to compare the half-lives of SARS-CoV-2 and a similar virus, SARS-Cov-1 in aerosol form and their surface stability on different materials. Mathematical models for the decay rate of each of each virus based on the decay rates observed in aerosols, copper, cardboard, stainless steel, and plastic. The half life of a viable concentration of the virus was largest on the surfaces of stainless steel and plastic. The half-life observed of SARS-CoV-2 (Coronavirus) on stainless steel and plaster were both close to 6 hours respectively. A viable virus was able to be detected up to 72 hours indicating fomite transmission of Coronavirus is especially plausible on the same materials used on a typical gas pump nozzle and buttons. The new nature of the Coronavirus pandemic leads the need to caution in concluding that viral transmission through gas stations is significant. The high traffic aspect of a gas station and single point of contact can only support the plausibility of a gas pump as a means of viral transmission.

Eliminating common points of aerosol and surface contact is the premise to the pragmatic approach of social distancing. When deterministic approach to remove the ways in which a virus such as Coronavirus can spread, it can be concluded that the virus will have a trend towards localized spread in various affected communities. The spread becomes slower when transmission is less likely from as many facets of transmission are eliminated. This method of preventative care has many supporting examples of this concept which can be applied to removing the common surface contact point of gas stations.

A study of South Korea's response to the Coronavirus pandemic supports the effectiveness of health guidelines that target limiting aerosol and fomite viral propagation [4]. Guidelines enforced in South Korea included the following:

- Extensive contact tracing
- Social distancing
- Wearing face masks
- Rapid and large scale testing
- Data to support contact tracing in the form of GPS, credit card transaction logs, and closed-circuit television

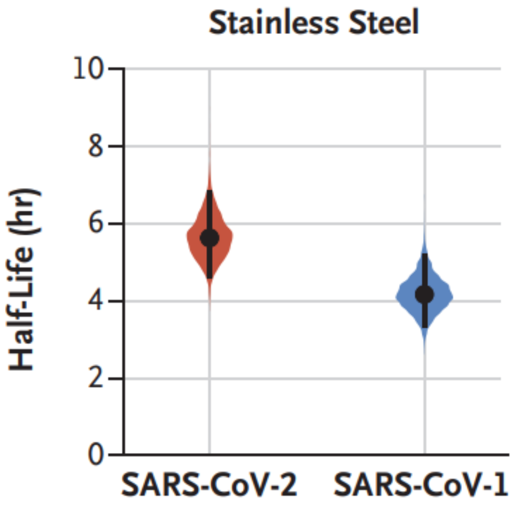


Fig. 10: Surface stability of SARS-CoV 1 and 2 on Plastic

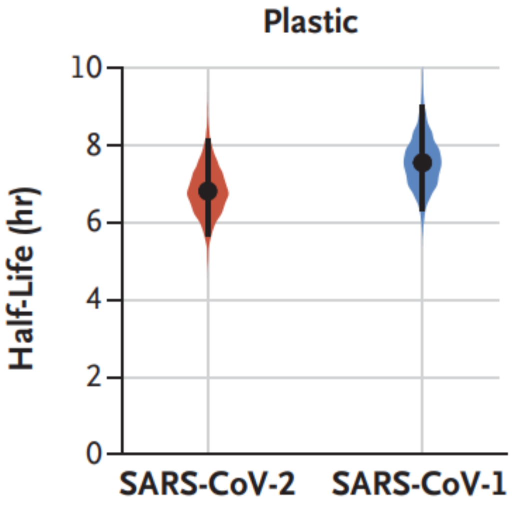


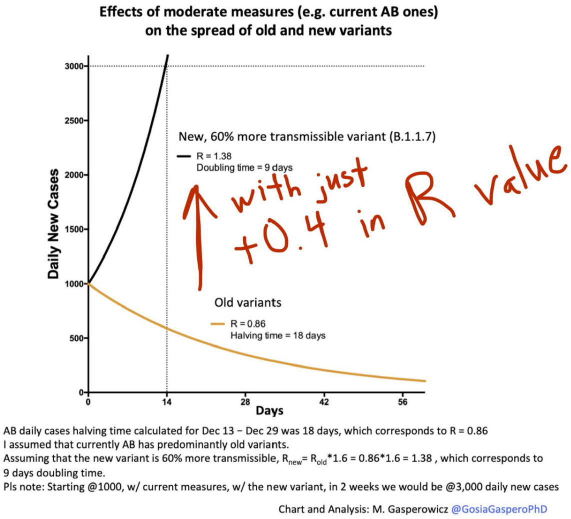
Fig. 11: Surface stability of SARS-CoV 1 and 2 on Plastic

The overall goals of South Korea's Coronavirus response was to use data driven means of determining areas of high congregation and implementing social distancing. With a high rate of public compliance to these measures, South Korea was able to rapidly lower the curve of infection and death rates of its population. The major upticks in viral transmission involved close contact at a megachurch, call centers, and various sporting event gatherings. The overall trend however, shows a steep trend to control viral transmission.

Trending data suggests that limiting person to person contact is an effective means to control the rate of infection in a population. The plausibility of viral transmission from common surfaces supports the possibility that fueling station pump handles are a potential issue that can be addressed in multiple ways. Solutions to lowering surface transmission at the pumps could include regular sanitizing, offering gloves for patrons to wear, or a full service option as seen in states such as New Jersey and Oregon. Regular sanitizing would ideally involve sanitizing between each customer use which would be logistically unfeasible. Public compliance with health orders or business level requirements for masks or glove usage is problematic given the current politicization to these types of requirements. A full service gas station poses a large financial burden to business owners due to the need for increased employees required for this solution. An a possible technological solution to the viral vector problem with gas pumps would involve removing the human aspect of pumping gas entirely.

There has been almost a year of changed practices regarding everyday life. Various parts of the globe have implemented virtual schooling, mandatory mask wearing, and rising levels of depression. This has been due to the COVID-19 virus and now there is evidence that this virus has changed for the worse. There has been one observed super strain as of late in the United States with another possibly on the way from South Africa. This new strain has been detected in six U.S. states; Colorado, Georgia, California, Pennsylvania and Florida. However, the recent spike as of 1/31/2021 is not attributed to these new strains. The danger of these newly found variants of the Coronavirus lies not within increased lethality but in the about 50 percent greater risk of transmission. "The danger of the new Covid-19 variant isn't that it's more severe, but that it's more contagious. If more people get infected this will increase the burden on already-overwhelmed hospitals—and lead to more deaths." -Dr. Tom Frieden. [25] The numbers don't appear to be reassuring. "Kucharski recently compared how many people would die after 10,000 new infections over the course of a month. With the current situation, with a virus that sees each patient infect an average of 1.1 others and kills .08 percent of everyone it infects, you'd predict 129 deaths over a month of spread. With a virus that's 50 percent more deadly, you would expect to see about 193 deaths over a month of spread. With a virus that's 50 percent more contagious, you end up with 978 more deaths over a month of spread – or five times as many deaths." [25] So unless the response to the pandemic on the part of the U.S. is altered the country could suffer catastrophic peaks of deaths and hospitalizations due to this so-called "super strain." It is still unclear on whether approved vaccines will still protect against such mutations to the virus, however, experts believe that due to the large immune response produced the new shots should still be effective. [25]

In the last 20 years there have been three significant pathogenic zoonotic disease outbreaks due to beta coronaviruses. These are SARS-CoV, which began in 2002 infecting around 8000 people with a 10 percent mortality rate, MERS-CoV, in 2012 with about 2300 cases and 35 percent mortality rate and the most recent SARS-CoV-2 which causes the severe respiratory disease COVID-19 with a reported 8.7 million cases and 460,000 deaths with increases daily. [26] When it comes to mutations, Coronaviruses have the ability to genetically proofread through inherent mechanisms which is why the diversity for the viral sequence is low. However, rare but favorable mutations can still occur due to natural selection. Most of the testing reagents for SARS-CoV-2 are based on the Spike protein sequence which mediates antibody-dependent enhancement (ADE) and worsens the disease in animal models. The most concerning variant is the U.K. discovered strain, B117, which has been given the nickname "beast mode COVID" is estimated to be 60 percent more transmissible than the initial strain observed in Wuhan China. [26] In Figure 12, the effects of current measures on the new and old strain variants can be observed. By implementing an automated gas pump the risk due to transmission through the touching of infected surfaces is greatly diminished which has growing importance as the virus mutates and becomes more transmissible.



Effects of moderate measures (e.g. current AB ones) on the spread of old and new variants

AB daily cases halving time calculated for Dec 13 – Dec 29 was 18 days, which corresponds to R = 0.86
I assumed that currently AB has predominantly old variants.
Assuming that the new variant is 60% more transmissible, $R_{new} = R_{old} * 1.6 = 0.86 * 1.6 = 1.38$, which corresponds to 9 days doubling time.
Pls note: Starting @1000, w/ current measures, w/ the new variant, in 2 weeks we would be @3,000 daily new cases

Chart and Analysis: M. Gasperowicz @GosiaGasperoPhD

Fig. 12: Effects of Moderate Measures on the Spread of Old and New Variants
[25]

F. Integration with electrical vehicles

As technology progresses electric vehicles (Evs) are becoming more readily available and cheaper to produce for the general public. These technological advances have caused a global interest in Evs as an alternative to vehicles power by fossil fuels as shown in Figure 12, which depicts the global increase in Ev stocks in the last few year, The increasing availability of electrical cars has caused many countries to petition for the transition to electric cars in place of gas and diesel vehicles, as a more environmentally friendly option. Countries such as Norway, France, the United Kingdom, India, China and more than 10 others have already announced their intentions to give up and get rid of gas and diesel powered cars by federally banning the sale of non-electric vehicles. Most of these bans are planned to be fully implemented within the next 2-3 decades. [8] Although the United States has not put any federal policies to get rid of gas and diesel vehicles, California Gov. Gavin Newsom has set similar bans into place for fossil fuel vehicles to greatly reduce carbon and greenhouse gas emissions in California. With more countries pledging to phase out gas and diesel powered vehicles to reduce carbon emissions, the increase in electric vehicles on the road will also increase the need for easily accessible charging stations.

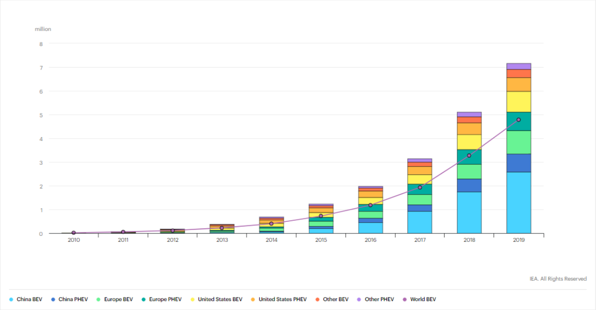


Fig. 13: Global electric car stock, 2010-2019 [9]

Integrating the automated aspects of the gas pump into charging stations for electric vehicles allows the same protection from payment fraud and disease transmission offered by the automated gas pump, while also accounting for the ever expanding market of electrical vehicles. Due to the country wide bans being put into on fossil fuel vehicles the standardization of electric vehicles is likely to take place in the next few years. As electrical vehicles become more standardized the frequency of use of charging station will increase, also increasing the need for the safety and efficiency offered by the automated charging stations and gas pumps.

G. *Infrastructure with autonomous vehicles*

Autonomous vehicles are the cutting-edge research topic ”du jour”, and test prototypes are making their way onto American roads. Companies like Google and Amazon are pouring money into their own research and development efforts, and private equity is lavishing cash on start-ups looking to find a niche in the growing industry.

The increasing availability of supporting technologies to automating the driving experience opens the door to associated infrastructure problems that need to be addressed. Autonomous vehicles will impact the way society uses and purchases vehicles. Taxi and ride sharing services could be completely driverless. Self driving cars could worsen gridlock in metropolitan areas with paid public parking. It is estimated that simply allowing autonomous cars to cruise at lower speeds in the city would cost between 29 to 50 cents per hour [6]. This estimate factors in energy use, depreciation, wear and tear, and maintenance. At a cruising cost of 29 to 50 cents per hour, it would be financially the better choice to allow these cars to cruise than pay for parking. This would add to the amount of traffic on the road and increase energy demands. A consumer who needs to refuel an autonomous electric car more often would need to account for the extra time needed to charge these vehicles. Rapid charging technologies are also emerging but at the cost of battery degradation. An autonomous charging infrastructure may benefit the consumer and public traffic conditions. Autonomous vehicle congestion can be lowered by offloading cruising time to charge and allow the consumer not to worry about the logistics of charging their car.

H. *Electric Vehicle Anxiety and Charging Infrastructure*

The adoption of electric vehicles has been slowed due to a multitude of factors. Two common factors include anxiety revolving around the range of electric vehicles and the accessibility of charging infrastructure. Regulatory and legislative solutions to these issues have been proposed to combat these issues. These policies anticipate that access to charging infrastructure will need to take into account the disparity of adoption to lower income communities. Apartments, lofts, student housing, and many forms of living accommodations likely do not have space or availability of a home charging solutions. This leads to the conclusion that people living in these forms of dwellings may not consider an electric vehicle as they become more accessible in price. Charging would be limited to charging at work or public charging stations. This lack of charging infrastructure could increase anxiety and thus the decision to use an electric car if it is believed that the range is insufficient (whether or not this is true).

The anxiety associated with the limited range of electric vehicles is a large psychological factor in the utility of owning an electric vehicles. A driver may choose not to use an electric vehicle for trips that are increasingly close to the estimated travel distance. This anxiety is warranted due to the highly variable nature of battery discharge. Factors such as temperature, acceleration, weather, or auxiliary systems for heating and cooling all effect the rate of discharge on a battery. This associated inaccuracy in estimating vehicle range can cause a driver to avoid a long trip if there is not a large enough margin for error in the indicated range of the vehicle.

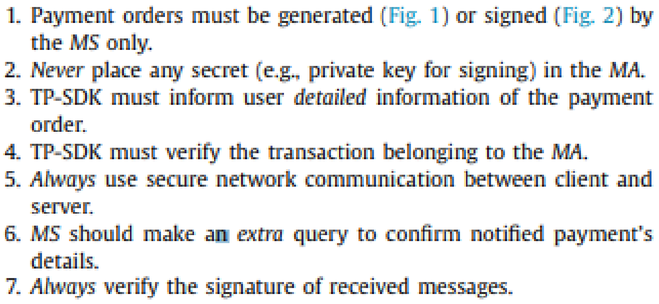
A simulation tool developed by the National Renewable Energy Laboratory gave way to a study simulating the impact of various electric vehicle charging infrastructure in relation to range anxiety[21]. The simulation tool accounted for battery chemistry, temperature, weather, acceleration, and many more factors to accurately study the life cycle of common electric vehicle batteries. Driving patterns collected from a traffic study were used to simulate the decision making of drivers if different forms of charging infrastructure was made available to them. A factor of range anxiety was included to study the utilization rate between charging options. Drivers could have high anxiety meaning they would not utilize the car for a trip if the range available was within 15 miles of the trip length. Drivers with low anxiety allowing for a 5 mile margin of error in the range were also studied. The largest driving utility found, was that of low power home charging. Adding workplace charging increased utility for long range commuters. Increasing the amount of charging options such as workplace charging and home charging was associated with lower range anxiety. In addition to the benefits of more charging availability, it was found battery degradation would be slower when vehicles are charged more often.

The adoption of electric vehicles is anticipated to be complex with many factors contributing to acceptance in various communities. The problem of range anxiety could be mitigated through widespread availability of charging infrastructure. Those in multi-unit dwellings would likely have the least utility since home charging is often unavailable. The study done associated with range anxiety did not account for lack of charger availability. The study assumed the chargers would be available and not occupied by other drivers. An automated fueling system could potentially be a solution to these problems given one of two things. The automated fueling system would either need to be mobile and charging different cars in their parking spots throughout the day or the vehicles would need to be autonomous and queue to the charging station throughout the day. Volkswagen has developed a solution to this problem that conforms to the former[22]. Volkswagen is providing this form of solution as one of the many ways to tackle the need for expanding charging infrastructure. This design is meant to integrate into restricted parking areas such as existing parking garages to allow early adoption. These two options could provide a means to increase electric vehicle adoption in the future.

I. Payment Security

As mobile technology improves many apps have incorporated different forms of payment with on of the most prominent being third party in-app payment. These third party payment systems often require communication with more servers and participants then traditional transactions. This makes transactions easier to exploit and more prone to error than other forms of transaction which can lead to serious financial in legal issues. With more and more apps centered around making it easier and faster for users to be able to pay for things like bills and being able to order the goods that they want directly to their houses.

There are many different processes for transactions in which cashiers of third party payment systems used to secure their transactions. Many successful third-party cashiers have developed specific security rules to prevent breaches in there systems.[23]



1. Payment orders must be generated (Fig. 1) or signed (Fig. 2) by the *MS* only.
2. *Never* place any secret (e.g., private key for signing) in the *MA*.
3. TP-SDK must inform user *detailed* information of the payment order.
4. TP-SDK must verify the transaction belonging to the *MA*.
5. *Always* use secure network communication between client and server.
6. *MS* should make an *extra* query to confirm notified payment's details.
7. *Always* verify the signature of received messages.

Fig. 14: 7 Security rules for mobile app payment [23]

If apps don't follow these rules it is likely that security breaches will take place. The insecurity of many in app payments is becoming one of the major threats to the mobile ecosystem that is evolving and including more online transactions. More often than not we find that apps often violate the security rules that are set out or are in properly implementing them leading to breaches in security. With the increase of mobile technology this foreseeing that many transactions that are done online or going to transition to apps rather than websites making it more important than ever to find sustainable insecure methods to conduct online transactions.

III. DESIGN IDEA

A. *Project Proposal Overview*

The proposed automated gas pump uses computer vision to detect a car, adjust robotic arm position to open the tank and gas cap, and insert the nozzle. The customer will park in a designated zone, pay via mobile app or NFC, and let the machine do the rest. This approach solves the various facets of our societal problem. The automated pump will eliminate spills and leaks due to topping off the tank, eliminate the manual gas nozzle as a vector for disease transmission, eliminate the possibility of credit card fraud through skimming, as well as integrate nicely with a fleet of autonomous vehicles.

This type of automated gas system does not exist in the United States - manual self-service stations have changed little since their introduction in 1964. Swedish engineering start-up Fuelmatics has proposed a similar autonomous system, though differing in execution - instead of an arm, the entire frame of the fuel station is moved to facilitate the insertion of a flexible hose and nozzle to the gas tank. It is also unclear how this design deals with the gas cap. This approach can be greatly simplified by using an arm instead of adjusting the position of an entire chassis.

B. *Technology Requirements*

This project will capitalize on an Intel Cyclone V(System on a Chip) architecture (see Appendix A for block diagram), aiming to greatly speed image processing using the inherent parallelism of FPGAs (Field Programmable Gate Arrays), combined with the flexibility of a connected embedded software system. SoCs greatly reduce design power consumption, space requirements, and cost. The specialized layout of these chips optimize data transmission by minimizing interference and interconnection delays. The FPGA will handle the large volume of image data in parallel, process it quickly and efficiently, then pass this data to the HPS(Hard Processor System) through the high frequency bridge that connects the two modules for further software processing. After processing, the HPS will pass control signals to the FPGA that will communicate with servos and actuators via digital and analog outputs. This specialized interaction between hardware and software components will optimize total system performance.

Changing a traditional gas pump to a full service gas pump that does not make use of a human attendant will require various forms of kinematic technologies. The kinematics of a robotic fuel station attendant will likely need to emulate movements that a human would normally be able to do. There is a large range of motion a gas station attendant or customer needs to be able to accomplish. The position in which a person drives into a fuel station is highly variable in their stationary position. The angle and translation of the car relative to the gas pump is different with each customer. The height of the vehicle is also variable relative to the gas pump. An ideal gas pump nozzle would therefore need to approach infinite degrees of freedom to accommodate for these factors. In this, the design of existing gas pumps is ideal since the nozzle positioning is limited mostly in the work space or reachable distance it is able to access. The positioning is not rigid and controlled by the high range of motion available through human dexterity. Replacing this system which has worked for generations will be no easy task and an adequate replacement must approach the kinematic ability that exists today with current gas pumps.

An actuation system that can provide the full range of motion existing gas pumps offer would be the ideal goal to achieve. Actuation to this degree would allow an automated gas pump to accommodate any vehicle type or positioning. This degree of actuation would also lend itself to practicality problems. Cost, complexity, size, speed, power consumption, and reliability could be problematic factors that would make this solution impractical. Limiting the range of motion the kinematic system is able to achieve could be the path to a viable automated gas pump. Targeting the most common vehicle types and positioning cases will guide a kinematic solution that can work for most users. According to the Department of Transportation, the two most common types of vehicles on the road are light duty vehicles with short or long wheel bases. They have accounted for close to 90 percent of highway vehicles registered since at least 2016. The fuel door on these types of vehicle are normally on either side of the vehicle with very few exceptions. Since there are no readily available statistics regarding how far away or at what angle a customer pulls into a fuel station, an estimate of a moderately sized and achievable working area will be assumed.

The kinematic capability of the system will accommodate a 3 dimensional reaching area of approximately 1000 millimeters of horizontal movement, 600 millimeters of vertical adjustment, and 1000 millimeters of reach. A minimum of three linear actuators is needed to accomplish this work area and allow dynamic reach for the translational variation of a light duty vehicle's position. A light duty vehicle fueling port is positioned at an angle for fuel to flow downward into the fuel tank. This angle must be accounted for and thus at minimum, one rotational axis is needed for this kinematic configuration. An initial prototype will therefore consist of three linear axes, and one rotational axis at the end effector. The three linear and one rotational axes are illustrated in Figures 13 and 14 with respect to the position of a conventional gas pump and vehicle position.

The technologies available for the kinematic system proposed can be achieved in a multitude of ways. A cost effective solution that can accommodate for the intended environment must be robust, minimal maintenance requirements, and simple in design. The horizontal axis can be achieved using a lead screw driven by a stepper motor to actuate the arm carriage horizontally. Mechanical limit switches can be used for position homing this axis offering a robust method of positioning. The vertical and end effector linear axes would ideally need absolute positioning so as not to require the fueling arm to protrude into the work area during a homing cycle. Absolute positioning would remove the need for a homing cycle on these axes. Conventional ball screw linear actuators with potentiometer position feedback would therefore be an adequate solution. The final axis, the rotational end effector axis, would also need absolute positioning and a minimum 90 degree amount of positioning. A small rotational servo motor would likely be able to perform the actuation necessary for this axis.

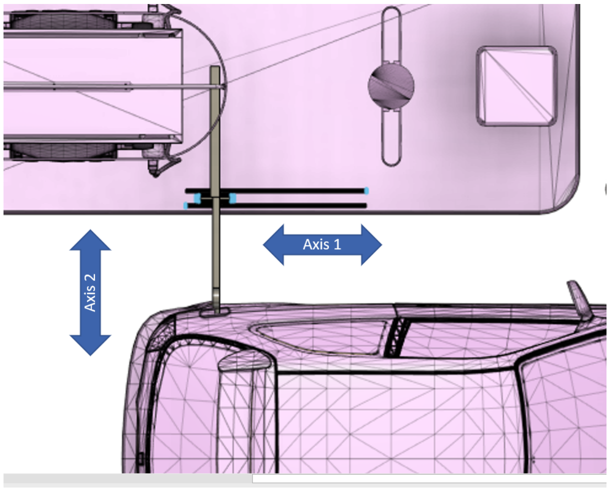


Fig. 15: Visual description of Axes 1 and 2

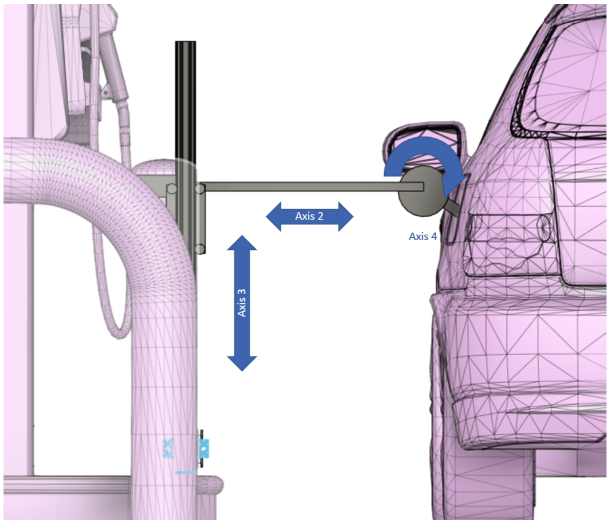


Fig. 16: Visual description of Axes 2, 3, and 4

The kinematic capability for the actuators needed in an automated fueling system requires position and velocity feedback. This is necessary in order to automate the process of fueling a vehicle. Accurate local and relative positioning is crucial in order to not damage vehicles using the system and so the system can properly plan its path to the fueling port. In addition to the local positioning mechanisms to the actuators driving each axis, ancillary feedback mechanisms can be used to provide better positioning feedback and redundancy to the system. An inertial measurement unit (IMU) provides angular and linear acceleration feedback. An IMU placed close or at the end effector of the protruding arm could allow more system stability if dual-loop control is implemented.

Simple limit switches on the nozzle end effector can be used to signal the robotics system that the nozzle has been inserted far enough into the tank. Used in combination with ultrasonic sensors, a fine-tuned final approach to the gas tank can be enabled.

Lidar is another option for generating a 3-dimensional point cloud. Lidar works by emitting lasers in a 360 degree pattern, and measuring how long it takes for those lasers to bounce back to the transceiver. This produces an extremely detailed 3-dimensional map with the lidar at the center.

Since the fuel station is stationary, there is no need to address portable power concerns. However, regulations regarding sufficient grounding near flammable liquids will need to be adhered to. 120V AC will be converted to various DC voltages($<$12 V) used to drive signals, sensors, actuators, and servos.

Remote Development Operations: Due to COVID-19, this project requires significant time investment and setup for a remote development environment. A VPN (Virtual Private Network) will be created so that all team members have access to a local, private network. A dedicated hardware server capable of running multiple VMs (Virtual Machines) will be present on the network. Each virtual machine will have a static IP assigned to it so that team members may access them via SSH (Secure Shell) or VNC (Virtual Network Computing). The server will be the host machine to which our programmable hardware will be connected (SoC). A webcam streaming through HTTP (HyperText Transfer Protocol) will be utilized for visual feedback from hardware programming. One guest VM will be reserved for programming hardware, and each team member will have their own personal guest VM, or 'sandbox', to develop and test code in a compartmentalized environment. Git will be utilized for version control. The developer will create a new branch for a particular feature. Once implementation, testing, and simulation of the new feature are complete, the developer will pull their 'build' of the source code into the dedicated hardware VM, and subsequently test it on the hardware. After confirming expected behavior, the developer will link a pull request to an issue on our GitHub repository, where it will be reviewed by all team members, then merged if it passes review. This ensures a stable environment for programming the hardware, keeping working directories clean and separate, as well as a pristine main branch at all times. Software-based diagnostic tools such as Digilent's Analog Discovery and Digital Discovery in conjunction with Waveforms will be used for hardware troubleshooting.

C. Features and measurable metrics

1) Kinematics: The required 4 degrees of freedom for this robotic arm imply 3 linear actuators to enable movement in the xyz plane, as well as one digital servo to control the angle of the nozzle. The feature will be considered complete when all degrees of freedom are achieved, as well as ensuring that the robotic arm is capable of the range of motion necessary to reach a car's gas tank and insert the nozzle.

The initial kinematic feature set will include the ability to position the fueling arm in three positions. The first position will be an idle or home position. This entails the fueling arm will be in a compact state not occupying the work area. The second position will be to traverse the arm to a reference position in front of a vehicle fueling port. This position will be a point in space relative to the fuel door which will be the origin point for the path planning to be executed for the final position. The final position will be the fueling position. This position is considered achieved when the fuel nozzle is inserted into the fuel port of a vehicle.

To achieve the three positions, a local positioning system must be implemented. This is the feature that will combine data from the motor feedback mechanisms and machine vision to allow for local positioning based on the position of a vehicle. This feature can be demonstrated by the ability to reach a reference position located in front of a vehicle fuel door. This feature set will be primary implemented by Ahmed Abdel-Gwad and is expected to require 300 hours to reach completion.

2) PID Control: The movement of the fueling arm must not cause damage to a vehicle. Proportional velocity and position control lends the risk of overshooting when moving to a specified position. Overshooting the movement of the fueling arm risks scratching or denting a vehicle while moving to one of the positions specified for fueling. Proportional integral derivative (PID) control will provide stability of movement by adjusting the signal response to position feedback and lowering the chance of a large response the closer the arm is to its target position. This feature can be measured by recording the proportional, integral, and derivative error measured while in use and the change in response as the arm reaches a target position or velocity. This will primarily be implemented by Steven Hellem using necessary skills from his control systems focus as well as Ahmed Abdel-Gwad. The forecasted number of hours for this is 200 hours.

3) Computer Vision: Machine vision and object detection play a large part in artificial intelligence and robotics today because it is how machines processes visual data. There are many different methods for image processing including a few different object detection algorithms. The most common object detection algorithms are Single Shot Detector (SSD), Region- based Convolutional Neural Networks(RCNN) , and You Only Look Once (YOLO)[16]. Traditionally RCNN and SSD are made to prioritize accuracy where as YOLO prioritizes speed. The object detection algorithms are used in conjunction with neural convolution networks and neural network algorithms like MobileNet to provide tracking along with the object detection.

MobileNet is a convolution neural network(CNN) model that is used for classifying images. MobileNet is not the only CNN architecture model that is used for machine vision and image classification. MobileNet is designed to be used for mobile devices and embedded systems[15]. Unlike many of the other CNN models MobileNet is optimized to run using very little computational power to apply machine learning. The central part of MobileNet is called, Depthwise Separable Convolution, are convolutions used to factorize a standard convolution into a depthwise convolution and a 1×1 convolution called a pointwise convolution [13]. Depthwise Separable Convolution splits into 2 layers, the first layer filters and the second layer combines the inputs into a new output set. This differs from standard convolutions, which filter and combine in one step.

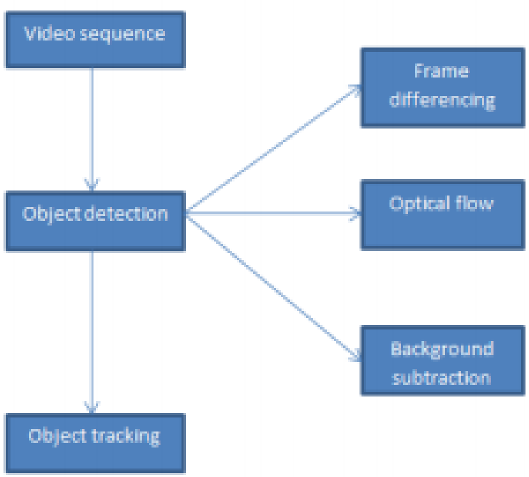
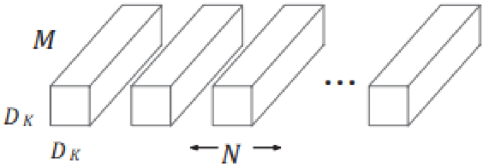
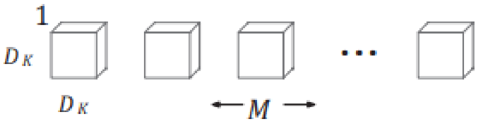


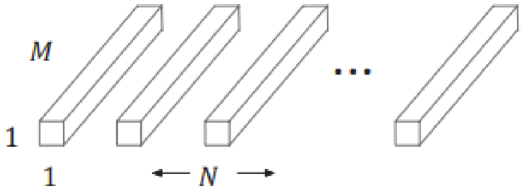
Fig. 17: Basic block diagram of detection and Tracking[16]



(a) StandardConvolutionFilters



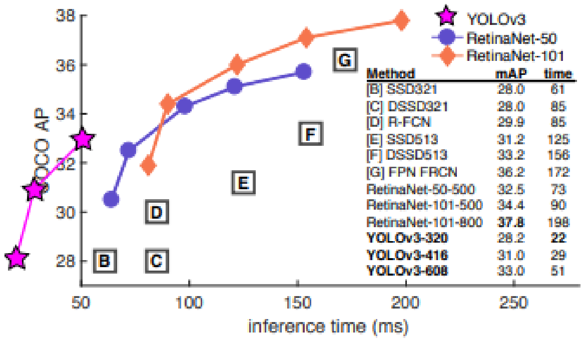
(b) Depthwise Convolutional Filters



(c) 1×1 Complex Filters named Pointwise Complexity in the context of Depthwise Distinguishable Complexity

Fig. 18: The standard convolution filters in (a) are substituted by two layers: depthwise convolution in (b) and pointwise convolution (c) to build a depthwise distinguishable filter[14].

YOLO is one of the fastest and most accurate object detection algorithms currently available. The YOLO algorithm works by applying a neural network to and entire image it then separates the image into different regions to predict the bounding boxes and probabilities of each region[17]. YOLO makes its predictions using a single neural network evaluation rather than using thousands for a single image like RCNN and Faster RCNN. This makes YOLO faster than most other object detection algorithms allowing it to run in real time. YOLO has been updated and reworked to provide more efficient object detection with its new version YOLOv3.



Method	mAP	time
[B] SSD321	28.0	61
[C] DSSD321	28.0	85
[D] R-FCN	29.9	85
[E] SSD513	31.2	125
[F] DSSD513	33.2	156
[G] FPN FRCN	36.2	172
RetinaNet-50-500	32.5	73
RetinaNet-101-500	34.4	90
RetinaNet-101-800	37.8	198
YOLOv3-320	28.2	**22**
YOLOv3-416	31.0	29
YOLOv3-608	33.0	51

Fig. 19: The performance speed for object detection algorithms. Showing that YOLOv3 runs significantly faster than other detection methods with comparable performance.[18]

YOLOv3 uses the darknet-53 network for feature extraction, while the previous version YOLOv2 uses the darknet-19 network[11]. Darknet-19 is a 30 layer architecture with a 19 layer network and 11 more layers for object detection. Darknet-53 is a 53 layer network with 53 more layers stacked onto it for object detection, for a total 106 layer convolution architecture[12].

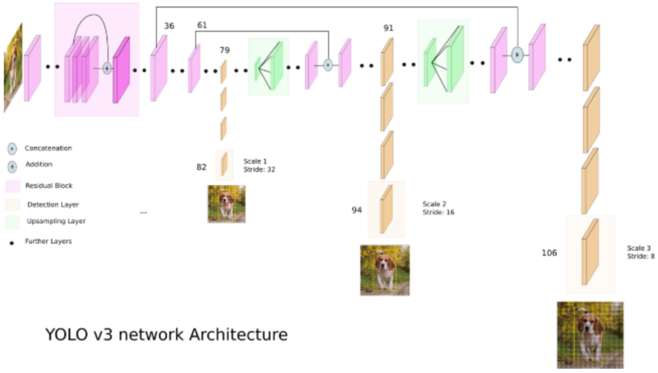


Fig. 20: Diagram showing the YOLOv3 architecture[12]

For sake of cost, a stereo camera will be used to gather RGB, depth data, and create a 3-dimensional point cloud of the camera frame. Because computer vision is a complex and cutting-edge subject, and in this scenario is the heart of our control system, this feature will be iterated on and improved as development proceeds. The first iteration will involve software image processing only. Future iterations will involve implementation of hardware-driven image processing in the FPGA. The team chose the Intel RealSense D435i camera because it integrates nicely with our Intel SoC, offers a full-fledged Software Development Kit(SDK) in the form of the Intel Realsense SDK 2.0, a well-documented API, and wrapper support for a variety of languages and frameworks, including C/C++, Python, Matlab, and OpenCV. The camera will be mounted stationary on the arm chassis, take pictures as the arm moves, and update the feedback loop. This feature will be considered done when the camera is capable of reliable informing the feedback control loop, and therefore guiding the movement of the robotic arm. A measurable metric for this feature would be outputting a bounding box location in the pixel grid of where the pump is, and also being able to distinguish whether a car is in the frame or not, and if that car is stationary or mobile. This is by far the most complex and time-consuming aspect of this project, and we estimate at least 400 hours to reach a "done" state. Michael Gomez and Jordan Moore are experienced C++ developers, and Jordan specifically has experience implementing computer vision with OpenCV. He will spearhead the machine vision effort, and Michael will provide support.

4) Mobile Application : The system must be capable of enabling mobile payments, backed up by NFC in case of app failure. This full-featured application must be capable of creating user accounts, securely transmitting payment information, giving directions to the nearest stations, and reporting current usage and waiting times. User account information will be securely stored in a remote database. The embedded system must support a connection to the internet, and receive a signal when payment is received to begin fueling. The app must be capable of interacting with the onboard GPS of the mobile device it is running on. Pump usage will be reported by the embedded system, and waiting times will be tracked with pressure sensors or cameras at the location of the fueling station. Because this is a true "full-stack" web application, modern development practices and frameworks will be utilized to facilitate rapid iteration. React Native is proposed as the front-end technology, as it is largely portable across Android and iOS devices, and supports JSX, a variation on JavaScript that supports inline CSS and HTML. The front-end will be coupled with a Flask back-end, written in Python. This back-end will handle interactions with the database and return serialized data to be represented by the front-end via REST API. The database proposed is MySQL, a robust solution with decades of documentation and corporate backing to its name. This application is estimated at 100 hours, as our developers are relatively inexperienced in developing mobile applications, and a significant amount of ramp-up is required for such a involved application. Michael Gomez has experience with developing web applications in Flask, and Jordan Moore has experience with HTML/CSS - they will be the principle mobile app developers. The feature will be considered fully implemented when customers are able to make a secure payment, the embedded system receives a signal to begin fueling after payment, it can give GPS-driven directions to the nearest station, and report usage and waiting times from each station. A measurable metric for this feature would be the ability to trigger fueling from a simulated "payment received" signal.

IV. FUNDING

TABLE I: Mechanical Bill of Materials

Received	Description	Quantity	Part Number	Price	Totals	Funding Source
x	MakerSlide (1000mm)	4	25142-11	24.99	99.96	Sponsor
x	Carriage Plate	5	25200-01	8.33	41.65	Sponsor
x	V-Wheel Kit (20 pk)	1	25203-08	84.99	84.99	Sponsor
x	MakerSlide (200mm)	1	25142-13	8.49	8.49	Sponsor
x	8mm x 1/2 shaft coupler	1		13.5	13.5	Sponsor
x	950mm lead screw	1	3501-0804-0950	20.99	20.99	Sponsor
x	8mm lead screw hub	2	545696	6.99	13.98	Sponsor
x	8mm lead screw nut	2	545315	7.99	15.98	Sponsor
x	[rgb] 1, 1, 08mm pillow block bearing	[rgb] 1, 1, 02	[rgb] 1, 1, 0535094	[rgb] 1, 1, 07.99	15.98	Sponsor

Inventables:Shipping Cost: $17.31
CA Sales Tax: $21.01

Total Cost: 315.52

Budget: 1500

TABLE II: Electrical Bill of Materials

Received	Description	Quantity	Part Number	Price
x	Linear Actuator (24” stroke)	1	3634	486.95
x	Intel Realsense D435i	1	D435i	199
x	2x20 Ribbon Cable (5pcs)	1		10.99
x	IDC40 Terminal Breakout	2		19.99
x	Linear Actuator (40” stroke)	1	PA-14P-40-50	158.99
x	DC Brushed Motor Controller	2	1363	39.95
x	12v 40A Power Supply	1		26.99
x	10x Micro Limit Switch	1		5.69

Total cost: 948.55

Budget: 1500

V. PROJECT MILESTONES

The significant milestones specific to the Robofuel project are outlined below in chronological order. These were used to gauge the progress of the design throughout the reporting period.

A. Build Progress Points

- Load ”golden hardware reference design” to FPGA
- Instantiated PID modules for motor control
- Used Qsys to define HPS-FPGA interface
- Implemented hardware accelerated image processing
- Generated preloader
- Genereated bootloader
- Generate device tree
- Built kernel
- Developed applications to run on custom kernel, and interact with FPGA IP
- Developed Linux application built on Intel Realsense SDK 2.0 to process image data and drive custom motor controls
- Developed web server to interact with mobile application
- Signaled robot arm to begin movement program after successful payment
- Created MySQL database to store user information
- Created a REST API using Flask to interact with database
- Reported streaming pump usage data to user
- Reported malfunctioning pump to user
- Integrated Google Maps API for easy directions to nearest station
- Integrated Flask back end with React Native front end
- Integrated Stripe API for secure payments
- Created a logo/banner
- Created a wireframe mockup of the application
- Useed React Native to create a cross-platform front-end
- Tuned and identified Gain values for optimization (Arduino)
- Implemented PID Controller on FPGA
- Created function to translate coordinates to position values for each actuator
- Built module to keep track of positions in system
- Created Stepper Motor homing/position tracker
- Created function to handle out of bounds position requests
- Created inverse kinematics function (generate actuator positioning from 3D coordinate input)
- Built function to pose at idle position
- Built function to pose in front of gas tank from coordinate generated from controller
- Calibrated Linear Actuator Feedback Devices
- Analyzed motor feedback signal and generated filter
- Evaluated range of travel signals and repeatability
- Built function to generate path to guide nozzle into gas tank
- Built function to generate shortest path to 3D coordinate
- Created test module for object detection using YOLOv3
- Used OpenCV to implement test module
- Calibrated OpenCV system
- Collected photographs of gashole
- Annotated gashole photographs for training
- Trained custom YOLOv3 module with gashole pictures
- Used OpenCV to test trained object detection module
- Calibrated the bounding boxes on detected object
- Intergrated Object Detection with Matlab
- Used bounding boxes on detected object to find center of gas hole
- Reported bounding box and gashole center to control through Matlab
- Intergrated depth data with object detection
- Translated computer vision data to kinematic positional data
- Calibrated camera placement
- Calibrated kinematic feedback based on machine vision data

VI. WORK BREAKDOWN STRUCTURE

A. Introduction

The project was organized using a work breakdown structure as described by the given course materials. The purpose of this section is to ensure the project continues forward correctly and abides by the rubrics provided. This also allows the team to project what is attainable in the given time frame as well as optimize our coordination of work packages in an asynchronous environment. Likewise, the team will be able to track progress on an individual level as well as adjust time-frames to remain on a forward trajectory.

Work to be Done

The work to be done can be divided into four distinct features. These include the embedded system, the mobile application, the pump kinematics, and computer vision. These features can be further divided into sub-tasks and work packages.

1) Embedded System: The embedded system can be divided into two components - the digital hardware design and implementation of embedded Linux. The digital hardware design will consist of four work packages. The first work package involves loading the "golden hardware reference design" (hereafter referred to as the GHRD) provided by Terasic for the DE-10 Nano development board to the FPGA. The GHRD provides a starting point for hardware designers, providing a reference that specifies parameters such as memory and clock speeds for a particular board. This will be one of the first tasks in the project scope, to be completed in the range of 16 Oct 2020 - 23 Oct 2020. After confirming that the GHRD is functional, the next work package will consist of instantiating a Verilog description of a PID motor controller, and integrating it with the GHRD. This will require use of Quartus Qsys (formerly known as the "Platform Designer"). Qsys will also define the HPS-FPGA interface, or "bridge", which will enable communication between the processor and our custom IP. Both these tasks are to be completed in the range of 30 Oct 2020 - 6 November 2020. Likewise, building a custom embedded Linux system will require several work packages, including generating a preloader, bootloader, device tree (which will include any custom hardware designs), and building the kernel itself. These steps should all be completed in the date range 23 October 2020 - 30 October 2020. After having constructed the embedded Linux kernel, a simple test user-space application will be written to confirm that the kernel includes our custom IP in the device tree. This will be completed in the range of 6 November 2020 to 13 November 2020. The final work package will consist of writing an application using the Intel Realsense SDK 2.0 to process image data, and use that processed data to drive the custom digital hardware. This final task will be completed within 13 November 2020 to 20 November 2020. Michael Gomez will be the principal engineer for the embedded system, and Ahmed Abdel-Gwad the secondary.

2) Mobile Application: The mobile application can be divided into two components - front-end and back-end. The majority of work-packages will be done on the back-end. These include integrating Stripe API for secure payments, the ability to interact with the internet-connected SOC to signal when to begin fueling, reporting pump usage data to users, as well as reporting any malfunctioning pumps at a given station, integrating the Google Maps API for quick direction to the nearest station, creating a MySQL database to store user information and pump data, creating a REST API with Flask for the front-end of the app to consume and interact with the database, and finally integrating our back-end with the front-end. Creating the front-end will require creating a logo or banner, a wireframe mockup of the app, and finally implementing it in JSX using React Native. The mobile application will be the focus of the second semester of the project. Michael Gomez will be the primary engineer, and Jordan Moore the secondary.

3) Kinematics: The kinematic feature set will be features critical to all movement in the automatic fuel pump. There will be five distinct features that make up the kinematics of the system. PID control, actuator local positioning, kinematic posing, position feedback, and path planning will be implemented to comprise the digital motion controls driven by the embedded system.

PID Control was selected for its proven ability to adjust for overshoot in an actuator. The motion of the automated fueling arm must not allow for damage to a vehicle. The work that will need to be done to implement this feature will revolve mostly around tuning the proportional, integral, and derivative responses of the controller. To allow the ability to develop the PID controller without the dependency of the embedded system implementation, a tuning and optimization test will be done on a common micro controller platform such as Arduino for parallel development. This work package will allow an in depth evaluation of interfacing with the two linear actuators with a PID controller and finding estimated values of ideal PID gains. This shall be completed between 18 Oct 2020 - 24 Oct 2020. The next work package will require creating a digital hardware design to be implemented on the FPGA part of the embedded system. The FPGA design shall consist of a digital hardware design equivalent of the micro controller test used for tuning. This work package shall have a hardware descriptive language (HDL) design with ports for input of a setpoint from the embedded system. This design is to be accomplished between 25 Oct 2020 - 31 Oct 2020. The final work package to complete the PID control system shall be a rapid tuning task of the digital hardware version of the PID controller using estimations provided form the first PID control work package. This will likely only take a day or two at most and should be accomplished the weekend of 31 Oct 2020 - 1 Nov 2020.

A local positioning function is necessary to keep track of the end effector position and the position of each actuator in the system. The first task to complete toward this would be to create a function to convert a 3D Cartesian coordinate to position values for each actuator in the system. This can be done in approximately a week on 1 Nov 2020 - 7 Nov 2020. A simple way to organize the state of the kinematic system would be to create registers or memory accessible to the embedded system that continuously keeps track of the current position of each actuator and end effector. This task can be completed within 6 Nov 2020 - 8 Nov 2020. The horizontal axis at the base of the fuel arm is to be controlled by a stepper motor and lead screw system which differs from the type of actuators used higher in the system. A task to control the stepper motor and keep track of positioning is required for the local positioning of this axis. A homing cycle must be created as a part of this function. A work package to create a digital control design for the stepper motor will likely take more than a week and can be completed between 8 Nov 2020 - 21 Nov 2020. The largest opportunity for errors may arise from a vehicle being in a position the fuel arm cannot reach. A function to the local positioning will include a fault reporting function to report if a commanded coordinate from the embedded system can be reached. This function will require the work area of the system is known and can be completed between 14 Nov 2020 - 15 Nov 2020.

The automatic fuel pump must be able to perform three distinct poses in the fueling process. Each of these poses has unique requirements and thus can be split into distinct tasks to complete. To accomplish each of these poses, an inverse kinematics function needs to be generated. This function will take a point in 3D space relative to the fueling arm and generate the rotational and linear setpoints needed to move the end effector to this point. This function shall be created within the week of 1 Nov 2020 - 7 Nov 2020. After implementing an inverse kinematics function specific to the automatic fuel arm degrees of freedom, an idle position is achievable. A function to pose the arm at a known idle position can be implemented within the weekend of 7 Nov 2020 - 8 Nov 2020. The second kinematic pose can be achieved concurrently to the idle position function. The second pose will be to position the fuel arm to a point center and directly in front of a vehicle fuel port. This function will be dependent on an estimated position provided by the machine vision functions and relayed by the embedded system. To allow a demonstration of this function without a dependency on the machine vision, this point can be measured manually and input to this function as an equivalent simulation of input. The final pose requires a path planning element to its functionality, therefore this work task will be considered under the path planning functionality.

Two linear actuators make up the primary axes of movement. These actuators have position feedback in the form of stationary potentiometers embedded in each actuator. This allows a simple form of absolute position feedback with the benefit of not having to perform a homing sequence upon startup. Position feedback for these motors will make up the fourth feature of the kinematics feature set. The first task to complete will be to calibrate the linear actuator feedback devices. This entails evaluating the sensitivity, range, and repeatability of these feedback signals. This task shall be completed within 11 Oct 2020 - 17 Oct 2020. The next task will be to analyze any noise apparent within the potentiometer signals, identify frequency ranges associated with the noise, and design a filter appropriate for this noise. This task shall be completed between 25 Oct 2020 - 31 Oct 2020. The final task necessary to complete the position feedback functionality will be to evaluate the range of travel corresponding to the linear actuator position feedback. This will entail identifying the signals measured across the range of travel and their repeatability. This task can be completed between 25 Oct 2020 - 31 Oct 2020.

The final feature to the kinematics feature set will be path planning. The first task to complete towards this feature will be to implement a function to generate the path that guides a nozzle into a vehicle fuel port. This feature requires the actuator to be in position 2, a known point relative to a vehicle fuel port. The path in its initial iteration will likely be a fixed path the end effector follows until in the final pose inside the fuel port. This task can likely be completed within two weeks between 22 Nov 2020 - 28 Nov 2020. The next task which is not critical to the system, will be to generate a shortest path to a given 3D coordinate. This will be a function that may be able to be completed within a week towards the end of November betweeen 29 Nov 2020 - 5 Dec 2020.

4) Computer Vision: The computer vision aspects of this project will we be controlling how how our robotic system will see and process the visual information that will be used for object detection.The object detection will be used in conjunction with the kinematics to provide positional feedback. The object detection system will use deep learning to be isolate specific regions on a live video feed so it can detect the specifically trained object, which in our case it is the fuel port on a car. Once the fuel port is detected it will provide this information to the kinematics system.

The computer vision section can be broken up into 2 sections, object detection and relative positioning. The object detection section has 8 work packages. The first work package involves creating a test module to train the YOLOv3 object detector using the darknet-53 framework. The test module will allow us to test out how we train the YOLOv3 system by using a Google provided data set so that we don't have to annotate and provide our own pictures for the test module. This will be completed during the week of 16 Oct. 2020 to 23 Oct. 2020. The next work package involves creating an integrated Python and C++ program with OpenCv in order to run the object detection on live video. This allows us to use our deep learning trained YOLOv3 system and to be able to apply it to live video for the purpose of conducting active object detection. This will be completed during the week of 16 Oct. 2020 to 23 Oct. 2020. After running the program with OpenCv the next step is to calibrate the program to make sure that the object detection for the training module properly functions when it is integrated. The completion range is from 23 Oct. 2020 to 30 Oct. 2020. Once the test module is running on the OpenCv system it will provide us with proof that object detection program works allowing us to move on to the training case that we are going to be using to detect the fuel port on the cars. The next work package involve collecting photographs that were going to need to train the YOLOv3 system to detect the fuel ports on the cars. The completion range is from 23 Oct. 2020 to 30 Oct. 2020. Once we have collected the images that we are going to need to train the the object detector for the fuel ports, we have to annotate the images. Annotating the images requires us to apply bounding boxes to the images that we have collected so that the YOLOv3 system can identify the specific regions of the photos in which we mean to detect.The completion range is from 30 Oct. 2020 to 6 Nov. 2020. The next work package involves training our own custom YOLOv3 module to detect the fuel port on specific cars. Training this module requires us to provide the YOLO system with the coordinates of the bounding boxes so that it can isolate the needed image regions before analyzing the RGB data to classify the object within the region. The completion range is from 30 Oct. 2020 to 6 Nov. 2020. The work package following this involves testing the OpenCv program with the custom trained module for still images and live video feed. This step is to detect if there are any errors in our object detection system that would stop us from properly integrating our object detector with our kinematics at a later time. The completion range is from 6 Nov. 2020 to 13 Nov. 2020. The final work package involves the calibration of the bounding boxes once they are detected on objects within the live feed. The regions for the bounding boxes on detected objects are going to be needed to provide the relative positioning in the next section. The completion range is from 6 Nov. 2020 to 13 Nov. 2020.

The relative positioning section focuses on using the data from the object detection system that we had previously built to integrate it with the kinematics to provide positional feedback based on the visual data that is taken from the camera. This section includes 7 work packages. The first work package involves integrating the object detection system with Matlab. We will be using Matlab to translate and generate the code for the kinematics on the FPGA system that we are using. The completion range is from 13 Nov. 2020 to 20 Nov. 2020. The next work package involves using the bounding box data that is provided from the trained object detector to calculate the center point of the fuel port so that we may set a point of insertion for our gas nozzle. The completion range is from 20 Nov. 2020 to 25 Nov. 2020. Once we find the point of insertion on the fuel port we can start the next work package which includes integrating the object detection data that we have with the depth data that is provided by the camera, to provide us with a 3-D representation of the space that we are viewing with the camera. The completion range is from 20 Nov. 2020 to 25 Nov. 2020. The following work package involves using the bounding box, fuel port center, and depth data to report this through Matlab to the FPGA system that will be using to control our kinematics. The next work package involves translating the computer vision data to kinematic positional data giving us positional feedback in addition the kinematic feedback data that is provided by the kinematic system. The next work package involves calibrating the camera placement on our system. To properly use the machine vision aspect with the kinematics the camera positioning has to be calibrated to find out where the best location on our system to provide the machine vision data to our kinematic system.The completion range is from 29 Nov. 2020 to 5 Dec. 2020.The next work package involves calibrating the kinematic feedback system to work using the machine vision data that we will provide from the object detection program. The completion range is from 29 Nov. 2020 to 5 Dec. 2020.

This information in tabular form can be found in Appendix G.

VII. RISK ASSESSMENT

This section will serve to identify possible associated risks with the team's project. This will include a description of each risk and mitigation techniques that have been and will continue to be utilized to ensure a safe and secure project. The risk assessment components are divided by key feature with extra sections for external considerations.

A. Embedded Systems

1) Associated Risks: This feature of the project requires a large amount of time and as implementing other features are dependent on this, there are risks of missing deadlines. The embedded system is the "heart" of the total system - it must function in order to drive the motors, run the image processing software, and facilitate communication between user and device. A stable embedded system is paramount to the success of this project - it must be capable of performing its prescribed tasks repetitively, and accurately. The goal is to enable 24 hour access to the pump, without requirement for human intervention. Because the SoC has limited processing power, it is possible that the hardware will not be capable of running the image processing software quickly enough.

2) Mitigation: The team decided to design and prototype the PID controller on an Arduino to identify roadblocks associated with the integration of the linear actuators. This also gave the team a general understanding of the behaviors of the actuators and provided insight to be used in the FPGA integration. After prototyping on the Arduino, the next step is porting the design to a custom FPGA block, and integrating it into an embedded Linux device tree. If the embedded SOC design does not work, the team will be able to fall back onto the Arduino implementation, or perhaps a more standard FPGA + microcontroller implementation. To mitigate the limited processing power of the SoC, the team will add an Intel Neural Compute Stick if necessary.

B. PID Control and Kinematics

1) Associated Risks: The Kinematics feature set encompasses the majority of high power electronics, mechanical, and electro-mechanical hardware. There majority of risks to completing this feature set would include a higher cost component failing or the time to replace a component not readily available. The Covid-19 pandemic has brought about delays in many supply chains across many industries.

The frame of the robotic fuel pump is designed around an open source design for an extruded aluminum v-rail. This design makes use of ball bearing v-wheels that may have issues involving the bearing assemblies coming loose, increase friction with dirt contamination on rails, and even break during operation which may damage the motors or equipment mounted to the frame.

The linear actuators used for the vertical and armature axes pose the highest risk if they fail. These actuators could pose a financial burden in the order of hundreds of dollars if they were to fail and needed to be replaced out of pocket. These actuators, if they fail, will also slow down the PID control development of the feature set. The motors are needed to tune the actuators based off of their feedback signals and physical behavior. The PID controller can be designed but the response gains would be difficult to accurately estimate without having the motors available.

An early goal in the selection of components for the kinematic system included choosing simple and enclosed components where possible. The linear actuators were chosen due to their strength as well as enclosed feedback components to allow for rapid prototyping. These feedback components are potentiometers that can provide an analog voltage signal proportional to the actuator shaft extension. Potentiometer feedback allows for an absolute position to be known without a homing cycle and immediately after startup. There would be no quadrature or sinusoidal encoder feedback to process which would allow more development time for the higher level features. Potentiometer feedback does pose a large risk to the design process. There is the possibility that the wiper can go through physical wear after some use. The wiper could output drifting voltage values in some areas of the turning range. This would produce erratic behavior in the PID controller if the potentiometer is degraded.

2) Mitigation: The associated risks outlined above can be mitigated by including necessary contingency plans or procedures if one of these scenarios were to occur. The original budget was closer to 3000 dollars. Careful selection of parts with this risk in mind allowed for an initial parts budget closer to half that figure. This allows room to lessen the financial burden associated with having to replace a high cost component out of the project budget. Parts were also selected based on their local availability. Any higher cost component was sourced from supplies preferably in California or the continental United States, with a year warranty where possible. The budget allows for a 100 percent cost buffer. Locally focused vendors allow minimal risk for shipping times and customer support.

The mechanical risks associated with the frame of the system are avoided by robust fastening. All bolts, bearings, and moving parts are inspected before operation. Bolts and bearing wheels are fastened with washers in combination with a thread lock fluid. Exposed moving components are lubricated and checked for dust, debris, and looseness.

Two linear actuators were chosen for two of the four axes driving the system. The reason two were chose was to have a similar device available if one were to fail. This would allow the development of the PID controller to continue as well as the ability to tune the controller for at least one of the actuators. The ability to continue development of this feature with a failed motor will allow the team to become familiar with the tuning process. When a replacement motor is acquired, the tuning process is known and a rough estimate of ideal values can be deduced from the alternate motor. If a replacement linear actuator is not able to be acquired, the horizontal stepper motor axis can be used as a backup alternate design. The horizontal axis at the base of the kinematic system uses a standard 3mm lead screw in combination with common bearings and a stepper motor. This is not ideal for one of the other linear axes but the components can be acquired from many sources and made to accommodate a missing linear actuator to achieve basic functionality.

The issues with potentiometer feedback are the largest concern in the stability of the kinematic system. The benefit of simple absolute positioning to the development process was decided that it would be worth the risk of it degrading over time. There are multiple ways to overcome this potential road block if it were to occur. The development process will make use of simulation where possible to minimize using the linear actuators continuously. The potentiometers used in the actuators are likely to fail before the actuator themselves, but will typically last for thousands or tens of thousands of cycles. The addition of secondary forms of feedback could also be used to overcome a road block such as this. The depth camera used for machine vision in the system has an embedded inertial measurement unit (IMU) built into the device. This camera can be mounted at the end effector as a backup form of position feedback. Acceleration feedback from the IMU can be used to confirm local positioning in the event the potentiometers degrade during the course of the project.

C. Machine Vision

1) Associated Risks: As the team's design is using machine vision for identifying positions for the gas pump to move, improper training of the incorporated object detection algorithm may lead to malfunctions in the gas pumps desired trajectory. As with improper PID tuning, this may lead to damage done to the customers vehicle and/or person. When dealing with the visual input that is required to conduct the machine vision there is the risk that non-ideal external conditions such as low lighting, could effect the system's ability to detect the gas port properly. Hardware can also pose another risk for the machine vision. Due to the limited processing power of the SoC it is possible that the object detection and image processing will not be able to run quickly enough. This can cause the kinematic system to be unable to properly guide itself which could also result in damage done to the customer's vehicle.

2) Mitigation: By including a large database of gas tank and car examples paired with proper training of the algorithm, malfunctions of the nature outlined above can be greatly mitigated. The issues posed by the PID tuning can be mitigated by proper calibration of the Machine vision and PID integration system.The calibration will properly correlate the image data we are receiving from the camera, with the kinematic feedback we are receiving from the PID controller in effort to improve kinematic accuracy and avoid damage to vehicles.

External conditions can pose issues with how the visual input is received. One of the ways that this can mitigated is by adding visual markers to the gas ports, such as qr codes, to allow the system to identify gas tank in spite of non-ideal external conditions. Another possible strategy to mitigate these issues would be to install a small light on or near the camera to get rid of issues like low lighting.

Once the machine vision and PID controls are integrated there is a possibility that the SoC will not have enough processing power to conducted the object detection quickly enough to actively interface with the kinematics. To mitigate this issue the team will add an Intel Neural Compute Stick. This will allow the system to have an increased amount of processing power without drastically increasing the amount of power needed to supply the system.

D. Mobile Application

1) Associated Risks: Considering our mobile application will be in part used for processing customer payments, there are security risks to the customer's personal information. This poses a liability risk to the team as well. The entire purpose of this system is a fully contact-less refueling experience - a mobile application is the only way to facilitate this.

2) Mitigation: Industry standard encryption will be implemented as part of the mobile application to secure user data. In case of mobile app failure, the design will include an NFC reader to maintain the contact-less payment system.

E. Covid-19 Pandemic/Federal and State Legislation

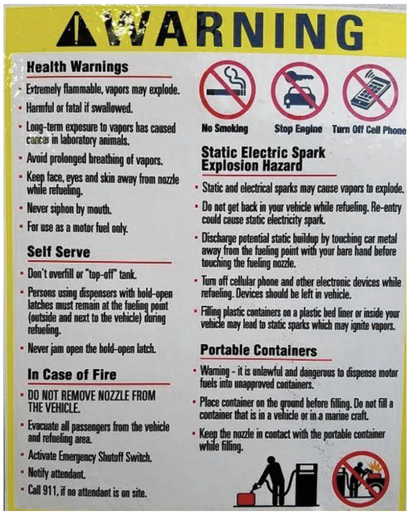
1) Associated Risks: During any pandemic there are inherent risks that need to be acknowledged. The first is risk of transmission of the virus by means of surface contamination in exchanging parts, close proximity interactions with team members and improper hygiene. There are also posed risks associated with compliance with local, state and federal guidelines. The CDC recommends that individuals be a minimum of 6 feet apart when interacting and wear masks. These risks are associated with testing of project equipment in person, exchanging of parts and any other person on person interactions required for efficient project development. Many of the same legislation surrounding gas stations will undoubtedly apply to our autonomous solution. This includes but is not limited to: only designated persons shall conduct fueling operations, engines shall be stopped during refueling, smoking and open flames shall be prohibited near refueling areas, equipment shall not be fueled or stored near underground entrances, elevator shafts or other places where gas or fumes might accumulate. [20] Warning signs associated with risks due to exposed fuel and vapors will also be implemented.

2) Mitigation: The associated risks outlined above can be mitigated in the following ways. The first is to make as many interactions as possible virtual. The team has set up virtual environments for meetings, programming and testing of machinery. The majority of parts were ordered to the host team member's house to prevent constant exchanges of equipment. When necessary person on person encounters happen, masks are worn and exchanged goods are sterilized.

/clearpage

VIII. DEVICE TEST PLAN

The following sections outline the various tests to be conducted from specific device tests through three tiers of integration. This is to ensure each device performs satisfactorily to the determined standard as well as integrates with



WARNING

Health Warnings

- Extremely flammable, vapors may explode.
- Harmful or fatal if swallowed.
- Long-term exposure to vapors has caused cancer in laboratory animals.
- Avoid prolonged breathing of vapors.
- Keep face, eyes and skin away from nozzle while refueling.
- Never siphon by mouth.
- For use as a motor fuel only.

Self Serve

- Don't overfill or "top-off" tank.
- Persons using dispensers with hold-open latches must remain at the fueling point (outside and next to the vehicle) during refueling.
- Never jam open the hold-open latch.

In Case of Fire

- DO NOT REMOVE NOZZLE FROM THE VEHICLE.
- Evacuate all passengers from the vehicle and refueling area.
- Activate Emergency Shutoff Switch.
- Notify attendant.
- Call 911, if no attendant is on site.

No Smoking
Stop Engine
Turn Off Cell Phone

Static Electric Spark Explosion Hazard

- Static and electrical sparks may cause vapors to explode.
- Do not get back in your vehicle while refueling. Re-entry could cause static electricity spark.
- Discharge potential static buildup by touching car metal away from the fueling point with your bare hand before touching the fueling nozzle.
- Turn off cellular phone and other electronic devices while refueling. Devices should be left in vehicle.
- Filling plastic containers on a plastic bed liner or inside your vehicle may lead to static sparks which may ignite vapors.

Portable Containers

- Warning - it is unlawful and dangerous to dispense motor fuels into unapproved containers.
- Place container on the ground before filling. Do not fill a container that is in a vehicle or in a marine craft.
- Keep the nozzle in contact with the portable container while filling.

Fig. 21: Gas Station Warning Disclaimer [19]

each other sub-component before testing the completed design. After each device is proven to be fully functional, they will be incrementally combined with their respective dependent devices to test for compatibility and expected functionality. The comprehensive list with accompanying dates can be found in Appendix I.

1) Device Tests: The first components to be tested are the linear actuators. The linear displacement will be within 2 mm at 10 different positions. This will be tested by putting in known coordinates and then measuring with a measuring tape to prove accuracy. The stepper motor tests will be conducted the same way. The FPGA/SOC will be tested for functionality and more specifically be proved to produce Linux boots, writing to the FPGA and can be accessed from the network. These three devices will be tested during the week of 2/22/2021-2/22/2021 by Ahmed Abdel-Gwad, Steven Hellem and Michael Gomez respectively.

2) Subsystem Tests: For the subsystems, 4 key components will be tested. These are the kinematics, the mobile application, the machine vision and the embedded system. Regarding the kinematics, the three-dimensional coordinate driven movement will be tested and must be within 4mm of the commanded position. This will be tested after the motors prove to be fully functional and accurate to the set requirements. The tests for this will be conducted by Steven Hellem in the week 3/8/2021-3/12/2021.

The mobile application will be tested by showing 100 percent unit test code coverage. It will be also be demonstrate the ability to store 10 user accounts. The mobile application will also be shown to be capable of gathering Google maps data. The mobile application testing will be conducted by Michael Gomez and Jordan Moore during the time period 3/22/2021-4/2/2021.

The embedded system will be tested by showing successful Verilog simulations, successful communication from the HPS and FPGA as well as 100 percent unit test code coverage. The embedded system will be jointly tested by Ahmed Abdel-Gwad and Michael Gomez during the time frame of 3/1/2021-3/12/2021.

The machine vision will be able to generate 3D coordinates of the fuel port. This will be tested by showing the fuel port machine vision generated location is within 2 cm of the true coordinates. The bounding box confidence will be shown by the software to be greater than 90 percent in 10 consecutive views. The machine vision will be jointly tested by Michael Gomez and Jordan Moore during the time period of 3/15/2021-3/26/2021.

3) 1st Tier Integration Test: Regarding the 1st Tier Integration Tests, the items to be tested are the kinematics and machine vision, mobile application and embedded system, and kinematics and embedded system. The kinematics and machine vision integration requires the machine vision's 3D coordinates to command the kinematics movement successfully 5-10 times consecutively. They must also be able to handle and deny out of bounds fuel ports. This integration test will be conducted by Ahmed Abdel-Gwad during the time period of 3/22/2021-4/2/2021.

The mobile application and embedded system integration must be able successfully communicate by receiving a command/status signal from each other 10 times consecutively. This integration will be tested by Jordan Moore during the week of 3/22/2021-3/26/2021.

The kinematics and embedded system integration will be tested by showing the 3D movement command was completed successfully 10 different points in space. The startup sequence also must be completed successfully 10 times with with the idle and position status saved in memory. This integration will be tested by Steven Hellem during the week of 3/22/2021-3/26/2021.

4) 2nd Tier Integration Test: For the 2nd Tier Integration Tests, the coordinates of all axis will be able to be generated. The coordinates of the camera are to be generated. Finally, 5-10 fueling sequences will be successfully completed. The three tests in this section will be conducted by Michael Gomez, Ahmed Abdel-Gwad and Steven respectively during the week of 3/29/2021-4/2/2021.

5) Complete Integration Test: After every subsystem/device is tested along with their respective integration combinations, the completed Robofuel system will be tested and will meet ready for market standards. The mobile application and autonomous fueling sequence will be shown to activate the fueling system 5-10 times consecutively. The system will be able to perform on more than 1 type of car successfully. Finally the autonomous fueling sequence will be shown to be fully functional in low light conditions. These final tests will be conducted by the entire team, Jordan Moore, Michael Gomez, Ahmed Abdel-Gwad, Steven Hellem respectively during the final week of testing, 3/29/2021-4/2/2021.

IX. DEPLOYABLE PROTOTYPE STATUS

A. Devices

1) Linear Actuators: The original system build utilized two axes driven by 24 inch and 40 inch linear actuators with built in potentiometers for position feedback. The measured size of a gas pump nozzle versus a gas tank port resulted in an outer diameter to inner diameter difference of close to 2 centimeters. An acceptable target for accuracy was determined to be approximately one tenth of the 2 centimeter figure for each axis of motion. This would provide a reasonable error budget of positional accuracy within 8 millimeters of 3D spacial accuracy. Initial test results from the first build shown in Tables III and IV show a high degree of inaccuracy within the linear actuators using potentiometer feedback. When calculated for an accuracy of 5 percent of expected resistance, the 24 inch actuator demonstrates an ideal accuracy of only 2 centimeters.

The linear actuators were upgraded to utilize dual quadrature encoders with a resolution of 360 pulses per revolution. The gearing that mated the original 10-turn potentiometers were adapted using 3D printed adapters as shown in Figures 22 and 23. The 24 inch actuator and encoder functioned correctly with the first assembly. The 40 inch actuator did not make use of the original 3D printed gearing similar to the one shown in Figure 23. This gear did skipped due to friction caused by inaccurate geometry from the 3D printed part. An alternative mounting solution was devised by boring out the encoder wheel to fit the larger gearing the original gear mated to as shown in Figure 22. The encoders were tested with results shown in Tables V and VI. Each actuator allowed for the target accuracy of 2 millimeters and was exceeded in most cases.

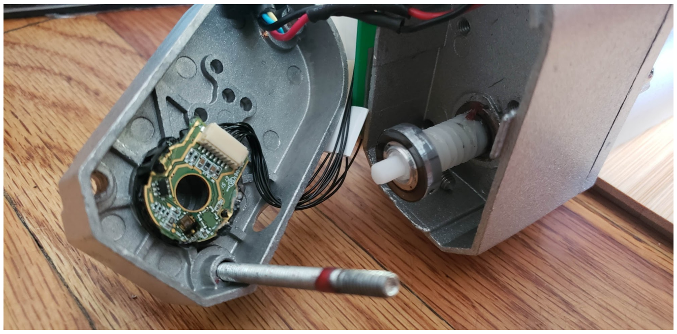


Fig. 22: 40 Inch Actuator Encoder Mounting

2) Stepper Motors: Initial tests of the stepper motors were close to the specified range given in the first device test plan despite the measuring method being less than optimal. After integrating the motor with the De-1 FPGA board, and using more consistent means of measurement, the results satisfied the requirement of 9 out of 10 commanded positions being within 2 mm of the expected positions. Both the motor and pillow bracket were secured to the chassis and the lead screw was greased to allow for optimal translation of the vertical axis attached carriage. The results can be observed in Table VII.

3) FPGA-SoC:

Fig. 23: 24 Inch Actuator 3D Printed Encoder Setup

TABLE III: 24 Inch Actuator Test With Potentiometer

Pos (cm)	Exp. Pos. (cm)	Pos. Diff. (cm)
10.5	12.76617718	-2.26618
13.3	15.52109819	-2.2211
17.2	19.47561638	-2.27562
20.5	22.79761735	-2.29762
23.5	25.76350599	-2.26351
26.6	28.87723909	-2.27724
31	33.34471331	-2.34471
35	37.22852327	-2.22852
40.7	43.07287947	-2.37288
45.2	47.54163929	-2.34164
49.6	51.97054538	-2.37055

TABLE IV: 24 Inch Actuator Test With Encoder

Pos (cm)	Exp. Pos. (cm)	Pos. Diff. (cm)
10	10.17861886	-0.17861886
289	286.6806109	2.31938911
122	122.1726428	-0.17264276
291	288.8240372	2.175962815
400	398.4960159	1.503984064
254	255.065073	-1.06507304
125.5	126.4594954	-0.95949535
58	58.04847278	-0.04847278
220	218.0909695	1.909030544
373	370.4528552	2.547144754

TABLE V: 40 Inch Actuator Test With Potentiometer

Pos (cm)	Exp. Pos. (cm)	Pos. Diff. (cm)
2.7	28.9122304	-26.2122304
5.4	32.39760744	-26.99760744
8.6	34.01949728	-25.41949728
11.5	34.05414317	-22.55414317
14.4	34.08555545	-19.68555545
16.7	34.08786518	-17.38786518
20.3	34.08555545	-13.78555545
23.7	34.0901749	-10.3901749
26.4	34.08601739	-7.686017395
27.7	34.09109879	-6.391098793
16.8	31.64740833	-14.84740833

TABLE VI: 40 Inch Actuator Test With Encoder

Pos (cm)	Exp. Pos. (cm)	Pos. Diff. (cm)
31	30.18363523	0.816364772
158	157.4790756	0.520924422
32	29.3482589	2.651741099
35	34.91743442	0.082565584
23.5	21.94921143	1.55078857
41	39.61145378	1.388546221
62	61.33123829	0.668761711
121.5	123.029747	-1.529747033
37	38.93519675	-1.935196752
50	49.07905215	0.920947845
97	96.17836508	0.821634916
118	117.4605715	0.539428482

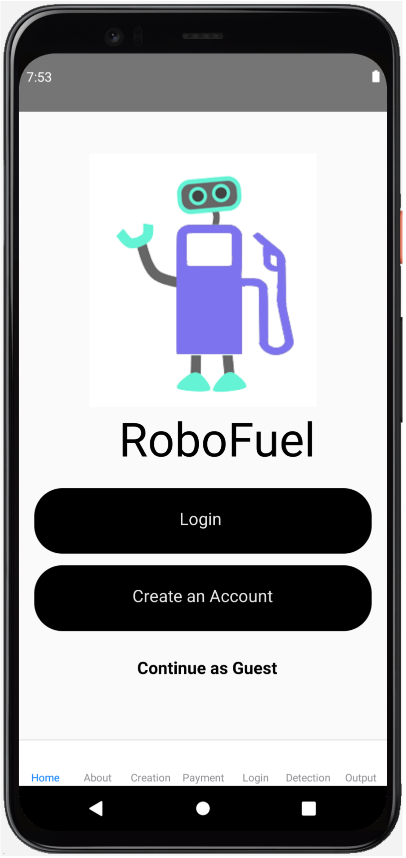


Fig. 24: Emulated Mobile App Home screen

B. Subsystems

1) Mobile App: Unit tests were written for the mobile app in Javascript, utilizing the test framework Mocha. This test framework allows for functional coverage, inspecting corner cases and insuring expected output against predetermined test cases. This gives proof of functionality on the software side, or client side. These tests were written such that there is 100% code coverage, and any new code must pass existing tests. This ensures robustness of the application. We also use an android emulator to simulate and test our app in real-time as it is developed.

2) Embedded System: The embedded system comes in two parts - customizable FPGA hardware and an embedded Linux system. Our team implemented custom motor controllers in Verilog, and those motor controllers were integrated into a custom device tree, part of U-Boot - the bootloader for our operating system. We were then able to test these custom hardware blocks by writing to memory-mapped addresses via the DevMem Linux utility - and reading back values that were manually written.

TABLE VII: Stepper Motor Position Test

Pos (mm)	Exp. Pos. (mm)	Pos. Diff. (mm)
77.5	78	-0.5
137.5	139	-1.5
156	158	-2
645	643	2
479	476	3
388	390	-2
450	452	-2
349	348	1
431	429	2
562	563	-1

To confirm the devices functionality for the intended use of communication between an FPGA and embedded Linux Operating system, 3 functional tasks needed to be demonstrated and observed. The FPGA must pass data to a program running on the embedded Linux part of the system, the embedded Linux should be able to map custom Verilog designs into memory, and the embedded software must be able to send data to the FPGA. To test and demonstrate this, a simple Verilog module was written to incorporate into the FPGA-SoC system. This module has two input ports and three output ports. The first input port takes a vector of inputs from the GPIO ports on the FPGA, passes the values through miscellaneous logic gates, then outputs the resulting value to an output port connected to LED indicators. The second output port connects the GPIO inputs to a port mapped to shared physical memory with the SoC. The same logic gate values are calculated in software and the resulting software output is written to the second input port of the Verilog design. The software calculated results are then displayed on LED indicators connected to the third output port to verify matching results and all three functional tests have passed. The values were changed 10 times to confirm repeated functionality without error.

TABLE VIII: 3D Coordinate Generation

3D Position (cm)	3D expected position (cm)	Diff
(105.8,-100,50)	(105,-100,50)	(0.8,0,0)
(55.4,-52.5,50)	(55,-52,50)	(0.8,0.5,0)
(110.5,–50.8,50)	(110,-50,50)	(0.5,0.8,0)
(90.6,-100.6,50.5)	(90,100,50)	(0.6,0.6,0.5)
(-110.7,-50,50.5)	(-110,-50,50)	(0.7,0,0.5)

3) Machine Vision: At first, our team was having issues with the accuracy of our de-projection from 2D pixel coordinates to 3D real world coordinates. After struggling with solving the problem in software, we realized that we needed to update our Intel Realsense camera's firmware, as well as run calibration software available from Intel. This enabled 3-dimensional accuracy to within 1cm cubed. The next major test was to verify the accuracy of the object detection algorithm that we trained. The results showed that as the video progressed the object detection accuracy improved more, with the detection accuracy over 90 percent per frame.

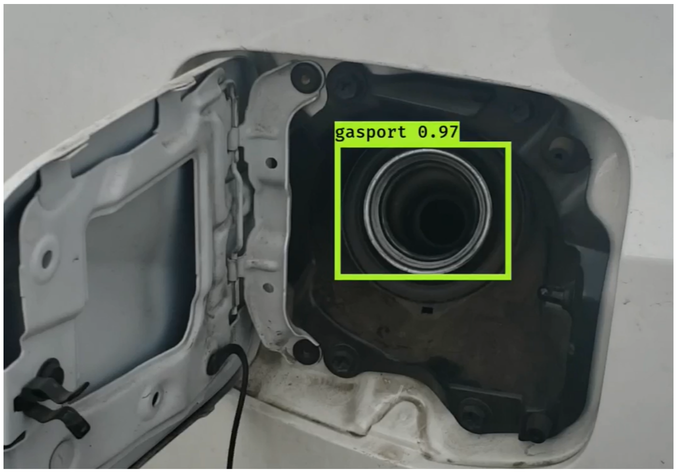


Fig. 25: Detected Fuel-port with Bounding Box

C. First Tier Integration

1) Kinematics & Machine Vision: The Intel Realsense SDK was used to generate 3D real world coordinates from 2D pixel coordinates of an image of the fuel-port with the bounding box surrounding it. These coordinates were used to inform the motor controllers.

2) Mobile App & Embedded System: The mobile app has a URL endpoint that triggers the machine vision running on our embedded system. This was done ten times repeatedly to insure reliability. This URL endpoint also fell under our unit testing scheme, guaranteed expected output with repetitive use and testing for corner cases.

D. 2nd Tier Integration Test

1) Kinematics & Machine Vision & Embedded System: The Intel Realsense SDK was used to deproject 2D pixel coordinates into 3D real world coordinates. These coordinates drove the motor controllers and insured correct positioning for the end effector of the device.

TABLE IX: Object Detection Test

Frame Number	Detection Accuracy
1	0.92
2	0.91
3	0.93
4	0.86
5	0.93
6	0.88
7	0.94
8	0.92
9	0.94
10	0.91
11	0.87
12	0.94
13	0.92
14	0.93
15	0.93
16	0.92
17	0.91
18	0.93
19	0.94
20	0.96
21	0.95
22	0.95
23	0.94
24	0.94
25	0.94
26	0.95
27	0.94
28	0.91
29	0.91
30	0.91
31	0.95
32	0.96
33	0.97
34	0.96
35	0.97
36	0.97
37	0.93
38	0.93
39	0.94
40	0.96
41	0.95
42	0.95
43	0.94
44	0.94
45	0.95
46	0.95
47	0.96
48	0.97
49	0.96
50	0.97

E. Complete Integration

The fully integrated design was the primary way to ultimately demonstrate all functions have been met. The completed integration allowed us to run the final tests of running complete fueling cycles under several conditions with the basic setup shown in Figure 24. The vehicle used for the training data set was used as the primary test target. This vehicle was placed in position in front of the fueling system with a 3D printed nozzle of the same size as a real nozzle attached to the end effector. The mobile app was then used to trigger a fueling cycle. The app sends a command over the internet, to trigger the machine vision system to calculate coordinates, the coordinates are then forwarded to the kinematic system, which confirms if the values are within the work area. The kinematic system then traverses to the valid coordinates from an idle position, to a reference position in front of the fuel port, and then an inserted position in the fuel port. The cycle is complete when the nozzle traverses back to the original idle position. This test was conducted 10 times to confirm reliable behavior. The same test was repeated in a low light condition in the evening, and with a different vehicle that was not part of the training images. Each test passed with exception for a 3rd vehicle that was used. The fuel port for the third vehicle was not recognized and would require additional images of the different fuel port for the training data set.

Fig. 26: Full Integration Test Setup

X. DESIGN PHILOSOPHY

The modern fuel station is a remnant of a centuries old technology which has made little improvement over the decades. Safety features have made pumps more accessible for self service but maintain flaws in their design making user error common. Fuel stations are rife with user error such as topping off which contaminates groundwater and air, not putting the nozzle back, and even the occasional fire from smoking. Fuel stations are also less accessible to some disabled customers when there are limited shop attendants. A disabled customer can find themselves in the situation of not having access to an attendant to help them with their fuel purchase. The emerging technologies of electric and self driving vehicles contain a host of new problems such as charging an autonomous vehicle or the lack of infrastructure available to owners. Our design aims to address all of these problems by developing an underlying technology for autonomous vehicle fueling.

Autonomous or semi-automated vehicle fueling has gone through many different development cycles through the decades by various entities. Various forms of solutions to the autonomous fueling problem have been attempted but not adopted for general use. The technologies have either been too expensive, complex, or impractical to use. Early mechanical solutions were not a one size fits all solution. Robotic solutions have come the closest to a practical solution that can support most vehicles but has always been an expensive endeavour. The modern expansion and availability of technology in the realm of machine vision, compact computing power, and re-programmable hardware make the potential adoption of autonomous fueling more realistic today. The motivation of applying this technology to emerging autonomous car technology further expands the need to make autonomous fueling a reality.

To tackle a realistic autonomous fueling platform, the barrier to entry must be low and reliability high. Our design attempts to cut the cost of an autonomous fueling system of the nearest competitor by a factor of ten. A simplified design utilizing reliable technologies to cater to the most common cases of vehicles are how this is achieved. The closest competitor uses a multitude of rotational joints to allow for a high degree of dexterity in order to emulate a human arm using a typical gas pump. Our design simplifies the solution down to the most necessary movements that need to be accomplished. The competitor design can be adapted as an add on to a traditional gas pump which offers an early adoption incentive. Our design can also be an add on unit to existing infrastructure. The design is simplified by incorporating a dedicated fuel nozzle to the system rather than utilizing the existing self-serve nozzles. This removes the need for a large degree of motion needed. When the nozzle is already a part of the fueling system, the only degrees of freedom needed are those that traverse the nozzle to the vehicle.

The movement from a fuel nozzle system to a standard side positioned vehicle fuel port was estimated to need three to five degrees of freedom depending on the portion of vehicles our system needed to accommodate. A minimum of three degrees of freedom is required to accommodate for the majority of vehicles with another two needed for motorcycles with a vertical fuel port and some older vehicles with a rear facing fuel port. The majority case utilizing the simplest design was the logical choice for designing a simplified and reliable system.

Three axes of linear motion allows for all movement necessary to fuel a vehicle. Two axes working synchronously allow for the insertion of a nozzle into a standard angled fuel port. All three axes will allow for a work space able to adjust for varying vehicle positions and angles. Lead and ball screw linear actuation have proven to allow for long term reliability with precise motion so as not to damage customer vehicles. This type of actuation allows for passive position stability which reduces power use rather than an active static positioning of a rotational joint based arm.

The design of the embedded control system and user interface maintained the drive towards low cost, simple, and reliable systems. An all in one depth mapping camera consolidates a normally two camera system into one allowing for less set up time and calibration. An FPGA SoC combination of devices allows for dedicated hardware to be deployed remotely with no need for physical setup. Client server software can be processed in a distributed manner on the same system reducing the need for a centralized server to handle all pumps. A simple mobile payment app removes the possibility of credit card skimmers commonly found on self service pumps and increases security for the customer.

XI. MARKETABILITY FORECAST

A. Industry Outlook

Automated refueling technology has been in development since the mid 1960s, starting with mechanical devices that required significant changes to the vehicle fill pipe, and evolving into modern automatons. Since 1990, there has been a significant renewed interest in automated refueling. Over 70 patents have been issued since then, covering all facets of robotic fueling, vehicle interface, instrumentation, and safety. Current developments in commercial refueling are being led by companies such as Shell Oil Company, Fuelmatics, Husky, Tokheim/Robosoft, Mercedes/BMW, and more.[27] Each of these designs are unique, but the fact that major companies are investing heavily into the concepts portends well for the future.

B. Potential Customers

This device would do well to target parents with young children or busy professionals hoping to save the time and bother of manually refueling their cars. Parents will no longer have to leave their children unattended, however momentarily, and their car can be refueled while they sit comfortably. Initially, this means targeting college-educated city-dwellers as a main demographic. Depending on reception, expansion into more diverse demographics as the device becomes more mainstream will encourage expansion.

In addition to gas stations, our device can be marketed and implemented by retail property managers or developers, rest stops and college campuses as a basic amenity. This is beneficial to both parties as the added convenience will attract new business as well as establish customer loyalty. The device should require minor modifications to be able to be compatible with existing electric vehicle charging stations.

Car dealerships and rental facilities deal with refueling their inventory on a daily basis. These places can be looked at for early integration of automated refuelling technologies. Dealerships and rental companies would be a good place to use for for early testing and integration of automated refueling systems because they deal with refueling large amounts of cars on a daily basis allowing for large scale testing before opening it up to a vast consumer market.

The integration of automated refueling systems can also help increase efficiency of businesses like car dealerships and rental companies. These systems can also allow another way to track the fuel use for each car used for rentals or test drives. Dealerships can also use on-site automated refueling to decrease liability. Dealerships without proper on-site refueling often have to sent there cars off the lot to get refueled which in turn increases fuel usage and increases chances of accidents causing inventory damage. With automated refueling systems these issues can be circumvented making car dealerships and rental companies the ideal customers for automated fueling systems like our own.

C. Competition

1) Fuelmatics:

The largest competitor in the commercial automated refueling space is a Swedish company called Fuelmatics. Fuelmatics has designed several iterations on a prototype refueling device, designed to work on regular consumer vehicles. No other company has filed more patents for the application of automatic refueling. Their design is very similar to ours in principle - customers will pay from a mobile app, which will trigger the kinematics system to facilitate fueling. They are using optical 3D pattern recognition technology to drive the kinematics. Though Fuelmatics is based out of Europe, they have an American division attempting to penetrate the market here as well. Their most recent prototype offers a substantially higher throughput for refueling. Refueling time from stop to go is cut by 30 - 50%. This means more customers per hour, or fewer pumps for the same amount of customers.

2) Scott Automation + Robotics:

Another similar system in the automated refueling space is Robofuel by Scott Automation + Robotics. The Scott Robotic refueling system mitigates the human risk involved with manned fuel tanks. Robofuel be placed “on the circuit” or “in-pit” allowing the the transportation time to and from fuel stations to be eliminated. Through the controlled pumping and coupling, the risk of environmental contamination due to spillages of toxic fuel particles are mitigated. Since this device eliminates on site personnel for refueling, those resources can be transferred to supervisory and confirmation tasks. Productivity is increased as the volume movement is increased. Along with the reduction of harmful hydrocarbons being exposed into the atmosphere and ground, risks associated with falling objects and machine failure are minimized. Robofuel is also configurable to accommodate various commercial fuel nozzles and receivers without the need for additional equipment to be added to the vehicle. The benefits of the automated refueling system as demonstrated by Scott Automation + Robotics are shared with our design. Our system, however, is targeted at consumer refueling. [28]

3) Airbus Defense and Space:

Companies like Airbus Defense and Space are on the cutting edge of in flight refueling technology. In 2017 the company showcased the technology in a demonstration of the world’s first automatic in flight refueling sequence between an aerial tanker and a F-16 supplied by the Portuguese Air Force. This demonstration lasted just over an hour at an altitude of 25000 feet. The process by which this feat is accomplished is first through the tracking of the receiver by the air refueling operator on the air tanker. Then image processing is used to determine the position of the refueling receptacle. There are multiple ways of controlling the telescoping beam inside of the boom. These are a manual mode by the operator, an assisted distance-keeping mode, and a fully automatic mode to make the contact. This technology will minimize workloads and be effect in less than optimal weather conditions. [29]

4) Autofuel:

Another major competitor in the field of automated refueling is Danish company Autofuel. Autofuel has designed a prototype that uses an AI assisted robotic arm system that can be added directly to fuel dispensers already in place at gas stations. This is another design that is similar to ours in principle. The AI robot is used to handle the 3D mapping of the car and process it using machine vision. The robotic arm system used has 3 separate arm functions, one is to open the gas port, the second is to remove the gas cap, and the third is to grab and pump the gas nozzle to dispense fuel. Autofuel uses programmable to RFID chip to handle its payments and customer information. The parking system is similar to that of a car wash with instruction appearing on the screen as they pull up.[30] This company looks like it is near the in its development phase and will mostly deployed in on some scale in Denmark within the next few years.

D. Industrial Fueling

Autonomous refueling of vehicles is a natural next step for the fuel service industry. Advancements and lower barrier to entry in robotics and stereo-vision have increased the likelihood of market receptiveness. An autonomous fueling system will however, be a slow adopter at the consumer level. The technology will need to be proven and reliable for almost all vehicles. The underlying technology of an autonomous fueling system will fortunately have industrial applications for early adoption.

An engineering group called Rotec Engineering has developed a robotic fueling system specifically for industrial applications [30]. The company targets a market with haul trucks, trains, straddle carriers, and more by means of efficient and autonomous refueling. Industrial and potentially remote operations could benefit greatly through increased safety for their workers and a more efficient fueling process.

Industrial processes are largely profitable from the scale and efficiency of all aspects of their operation. The largest profit generation is normally the ability to process, manufacture, or service a product in the least amount of time to maximize throughput. Industries such as mining, oil and gas, or construction would be the ideal early market for an autonomous refueling system. Vehicles are often of similar type and size making adapting the system to a specific consumer a quick process. The system can be adapted to be mobile for remote unmanned refueling locations. Operators are removed from the process of fueling, thus a quicker process can be achieved by means of robotic actuation. A more efficient refueling process creates efficiency for industrial operations.

E. Market Advantages of Robofuel

The primary advantage of the proposed Robofuel design is in relation to cost. Utilizing reliable and low cost technologies puts the Robofuel design at a decided advantage. The emerging market of automated refueling has shown the potential for multi-industry applications with little to no modifications. The system prototype was developed with approximately 1800 work-hours split between four engineers. The research and development cost amounts to a small fraction in comparison to the hours companies such as Husky, Shell, or car manufacturers have devoted. The Robofuel system offers an earlier profitability timeline when considering the development costs.

Many competitors within the automated refueling market have only reached the prototype stage which leaves a one to one cost comparison unclear. The closest comparison of a similar system can be found with the Husky and Fuelmatics partnered design. The design is estimated to cost $50,000 for each fueling unit [32]. The Robofuel deployable prototype has a materials cost of under $2,000 with an estimated total cost including work-hours of $10,000. There are no licensing fees associated with the software used in the design. Underlying software utilizes open-source tools to keep costs relatively fixed. The Robofuel design has a market advantage of being an order of magnitude lower in cost compared to the closest market competitor.

XII. CONCLUSION

The recent Coronavirus pandemic has illustrated the increased need to address the societal problem of today's gas pumps. Current gas pumps (and similarly vehicle charging stations) pose an environmental, accessibility, and public health risk to society. The modern fueling station needs to be reassessed to address the multitude of issues with how we fuel our vehicles. Environmental contamination in the form of groundwater contamination can affect wildlife and humans. Carcinogens such as benzene are apparent in gasoline and dropped in large quantities over time through the misuse of gas pumps. Disabled drivers in the United States face issues with accessibility and a lack of attendants able to assist disabled drivers could be addressed. In addition to the environmental health impacts caused from the misuse of gas pumps, the materials gas pumps are made of may prove to be a viral vector of transmission. Many of the issues listed revolve around one problem with the modern fueling station. The human element of operating a fuel pump causes many problems that can be addressed by removing human operation from the equation. It can be argued a change to full-service fueling may be the best solution to the problem with the modern gas station. The financial burden of implementing this with trained attendants may be a non-starter to a solution. A robotic full-service station may be a feasible financial alternative that could also address the issues with self-service stations.

The human element of operating a fuel pump causes many problems that can be addressed by removing human operation from the equation. It can be argued a change to full-service fueling may be the best solution to the problem with the modern gas station. The financial burden of implementing this with trained attendants may be a non-starter to a solution. A robotic full-service station may be a feasible financial alternative that could also address the issues with self-service stations.

To address these problems, a design based on computer vision and a robotic arm with 4 degrees of freedom will be implemented to facilitate an automated gas pump. The system will take advantage of a System-On-Chip architecture, which includes an FPGA and computer processor on the same chip. This means that the design will take advantage of the parallel nature of the FPGA, rapidly processing image data, the high speed data transmission between FPGA and CPU, and the flexibility of a software system for controls and further processing.

The operation of the design will be as follows: First the automobile will be driven to a predetermined and marked location next to the gas pump. The refueling will start upon confirmation through the phone application by the customer. The computer vision will locate and send information to the FPGA allowing the robotic arm to link the gas hole to the nozzle. Fuelling will then commence and post-billing information will be handled through the user's application.

GLOSSARY

SARS-Cov-2: Also called severe acute respiratory syndrome coronavirus 2. The virus that causes a respiratory disease called coronavirus disease 19 (COVID-19). SARS-CoV-2 is a member of a large family of viruses called coronaviruses.

coronavirus: any of a group of RNA viruses that cause a variety of diseases in humans and other animals.

hydrocarbon: a compound of hydrogen and carbon, such as any of those which are the chief components of petroleum and natural gas.

fomites: objects or materials which are likely to carry infection, such as clothes, utensils, and furniture.

NFC (Near Field Communication): NFC is a short-range high frequency wireless communication technology that enables the exchange of data between devices over about a 10 cm distance.

FPGA: A field-programmable gate array is an integrated circuit designed to be configured by a customer or a designer after manufacturing – hence the term ”field-programmable”.

SOC: A system on a chip is an integrated circuit (also known as a ”chip”) that integrates all or most components of a computer or other electronic system.

Kinematics: The branch of mechanics concerned with the motion of objects without reference to the forces which cause the motion.

REFERENCES

[1] Begian, Charles, and Houssain Kettani. ”Analysis of Fuel Pump Skimming Devices.” Proceedings of the 2020 the 4th International Conference on Information System and Data Mining. 2020.

[2] Bass, Sarah Bauerle, et al. ”If you ask them, will they come? Predictors of quarantine compliance during a hypothetical avian influenza pandemic: Results from a statewide survey.” Disaster medicine and public health preparedness 4.2 (2010): 135-144.

[3] Cava, Maureen A., et al. ”Risk perception and compliance with quarantine during the SARS outbreak.” Journal of Nursing Scholarship 37.4 (2005): 343-347.

[4] Kim, Eun A. ”Social Distancing and Public Health Guidelines at Workplaces in Korea: Responses to Coronavirus Disease-19.” Safety and Health at Work (2020).

[5] Van Doremalen, Neeltje, et al. ”Aerosol and surface stability of SARS-CoV-2 as compared with SARS-CoV-1.” New England Journal of Medicine 382.16 (2020): 1564-1567.

[6] Millard-Ball, Adam. ”The autonomous vehicle parking problem.” Transport Policy 75 (2019): 99-108.

[7] Hilpert, Markus, and Patrick N. Breysse. ”Infiltration and evaporation of small hydrocarbon spills at gas stations.” Journal of contaminant hydrology 170 (2014): 39-52.

[8] Kass, Madeline J. ”The End of the Road for Gas-Powered Automobiles?” Natural Resources Environment, vol. 32, no. 4, 2018, pp. 53-54.

[9] Iea.“Global EV Outlook 2020 – Analysis.” IEA

[10] Geraldino, Barbara Rodrigues, et al. ”Analysis of Benzene Exposure in Gas Station Workers Using Trans, Trans-Muconic Acid.” International Journal of Environmental Research and Public Health 17.15 (2020): 5295.

[11] J. Hu, X. Gao, H. Wu and S. Gao, ”Detection of Workers Without the Helments in Videos Based on YOLO V3,” 2019 12th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Suzhou, China, 2019, pp. 1-4.

[12] Kathuria, Ayoosh. “What’s New in YOLO v3?” Medium, Towards Data Science, 29 Apr. 2018

[13] Howard, Andrew G., et al. ”Mobilenets: Efficient convolutional neural networks for mobile vision applications.” arXiv preprint arXiv:1704.04861 (2017).

[14] N. R. Gavai, Y. A. Jakhade, S. A. Tribhuvan and R. Bhattad, ”MobileNets for flower classification using TensorFlow,” 2017 International Conference on Big Data, IoT and Data Science (BID), Pune, 2017, pp. 154-158

[15] Pradhan, Mithilesh. “Image Classification Using MobileNet in the Browser.” Medium, Analytics Vidhya, 1 Dec. 2019

[16] G. Chandan, A. Jain, H. Jain and Mohana, ”Real Time Object Detection and Tracking Using Deep Learning and OpenCV,” 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, 2018, pp. 1305-1308

[17] Science, ODSC - Open Data. “Overview of the YOLO Object Detection Algorithm.” Medium, Medium, 25 Sept. 2018

[18] Redmon, J. and Farhadi, A., 2018. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767(2018).

[19] Dubin, S., 2020. Column: Lessons From The Warning Signs At Gas Stations. [online] chicagotribune.com. Available at: ¡https://www.chicagotribune.com/ct-bgc-gas-safety-col-tl-1208-20161201-column.html¿ [Accessed 9 November 2020].

[20] LII / Legal Information Institute. 2020. 29 CFR § 1917.156 - Fuel Handling And Storage.. [online] Available at: ¡https://www.law.cornell.edu/cfr/text/29/1917.156¿ [Accessed 9 November 2020].

[21] Neubauer, Jeremy, and Wood, Eric. “The Impact of Range Anxiety and Home, Workplace, and Public Charging Infrastructure on Simulated Battery Electric Vehicle Lifetime Utility.” Journal of Power Sources, vol. 257, 2014, pp. 12–20.

[22] “Volkswagen Group Previews Mobile Charging Robot.” Volkswagen US Media Site, Volkswagen Group, 28 Dec. 2020, media.vw.com/en-us/releases/1452.

[23] Yang, Wenbo, et al. “Security Analysis of Third-Party in-App Payment in Mobile Applications.” Journal of Information Security and Applications, Elsevier, 23 July 2019

[24] Klosowski, Thorin. “Which Mobile Payments Are the Most Private and Secure?” The New York Times, The New York Times, 2020

[25] Goodman, Brenda. “New COVID ’Su-per Strains’ Could Disrupt Life Again.” WebMD, WebMD, 6Jan. 2021, www.webmd.com/lung/news/20210107/new-covid-super-strains-could-disrupt-life-again.

[26] Korber, Bette, et al. ”Tracking changes inSARS-CoV-2 Spike: evidence that D614G increases infectivity of the COVID-19 virus.” Cell 182.4 (2020): 812-827.

[27] ”Is The Long Wait for Robotic Fueling Finally Over?” (PDF). Wolf H. Koch, Ph.D. September 1997. Retrieved October 16, 2014.

[28] Scott. “Robofuel.” Scott, Scott Automation, www.scottautomation.com/products/robofuel/.

[29] Staff Writer. “Airbus Demos New Automated Aerial Refueling Technology.” Aviation Today, Aviation Today, 12 May 2017, www.aviationtoday.com/2017/05/12/airbus-demonstrates-new-automated-aerial-refueling-technology/.

[30] “Robotic Fuelling System for Haul Trucks ,Mining Trucks , Trains and AGVs.” Rotec Engineering, 26 May 2020, rotec-engineering.nl/robotic-fuelling-systems/robotic-fuelling-system-for-haul-trucks/toggle-id-1.

[31] “Home: Autofuel - Automatic Refueling. For Your Convenience.” Autofuel, 2018, www.autofuel.eu/.

[32] Main, Douglas. “Robotic Gas Pumps Are Coming Soon.” Pop-

ular Science, www.popsci.com/article/cars/robotic-gas-pumps-are-coming-soon/.

[33] “Depth Camera D435.” Intel® RealSense™ Depth and Tracking Cameras, 23 Mar. 2021, www.intelrealsense.com/depth-camera-d435/.

User Manual

1.) Pull Car Up To The Line Designated At The Fueling Station
2.) Open Up Your Fuel Port
3.) Open Mobile App
4.) Login Or Sign Up If You Don't Have An Account
5.) Select Payment Method
6.) Choose Desired Fueling Amount
7.) Begin Fueling
8.) Wait For Fueling To Finish
9.) Close Your Fuel Port

Mobile App Overview

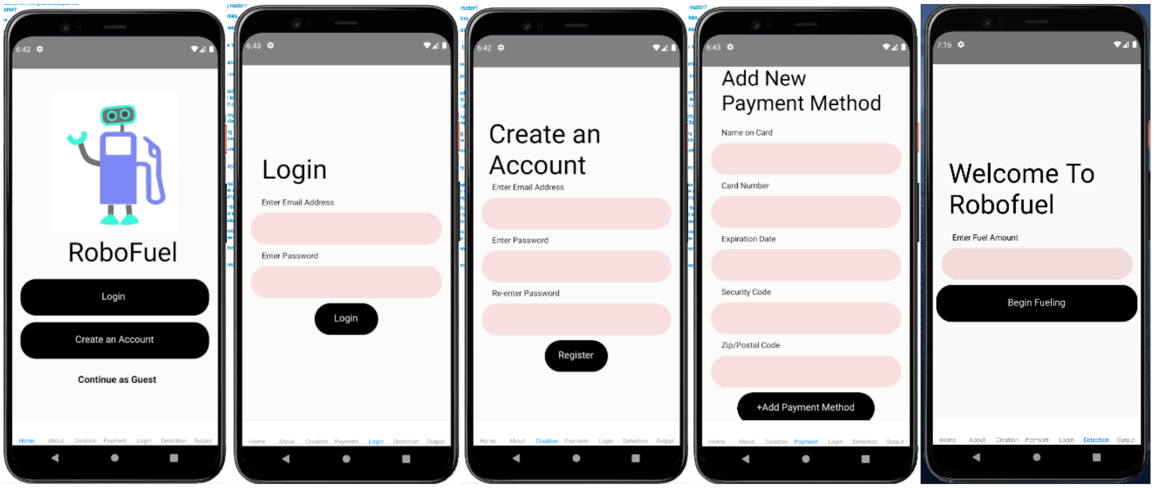


Figure 1: Step 3 *Figure 2: Step 4* *Figure 3: Step 4* *Figure 4: Step 5* *Figure 5: Step 6 & 7*

Fig. 27: User Manual

XIV. APPENDIX B: HARDWARE

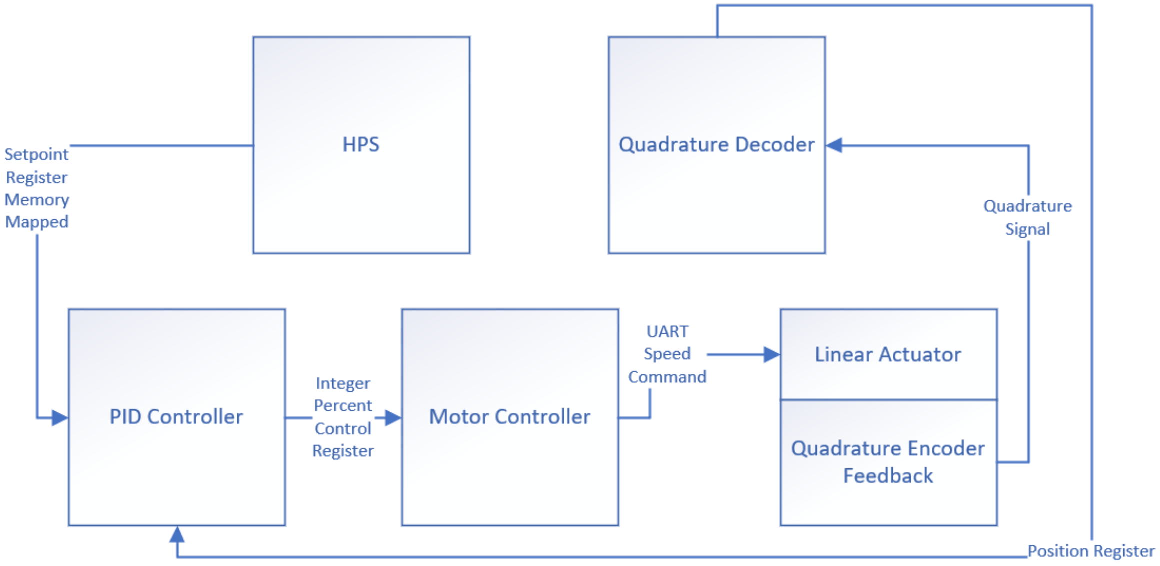


Fig. 28: Hardware Block Diagram

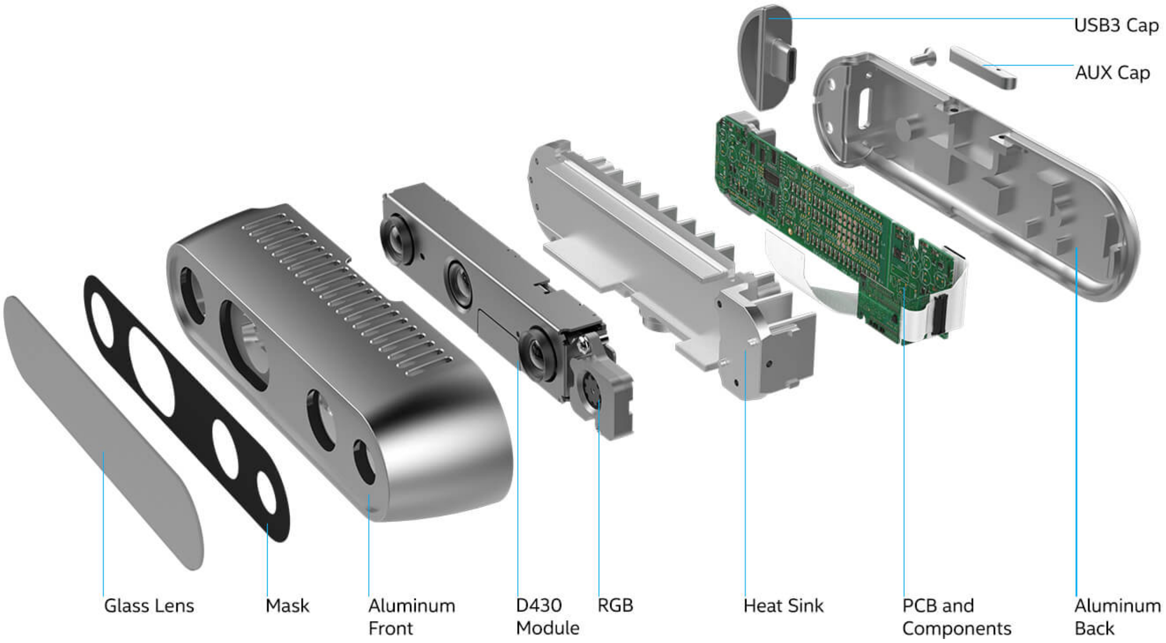


Fig. 29: Camera Schematic

XV. APPENDIX C: SOFTWARE

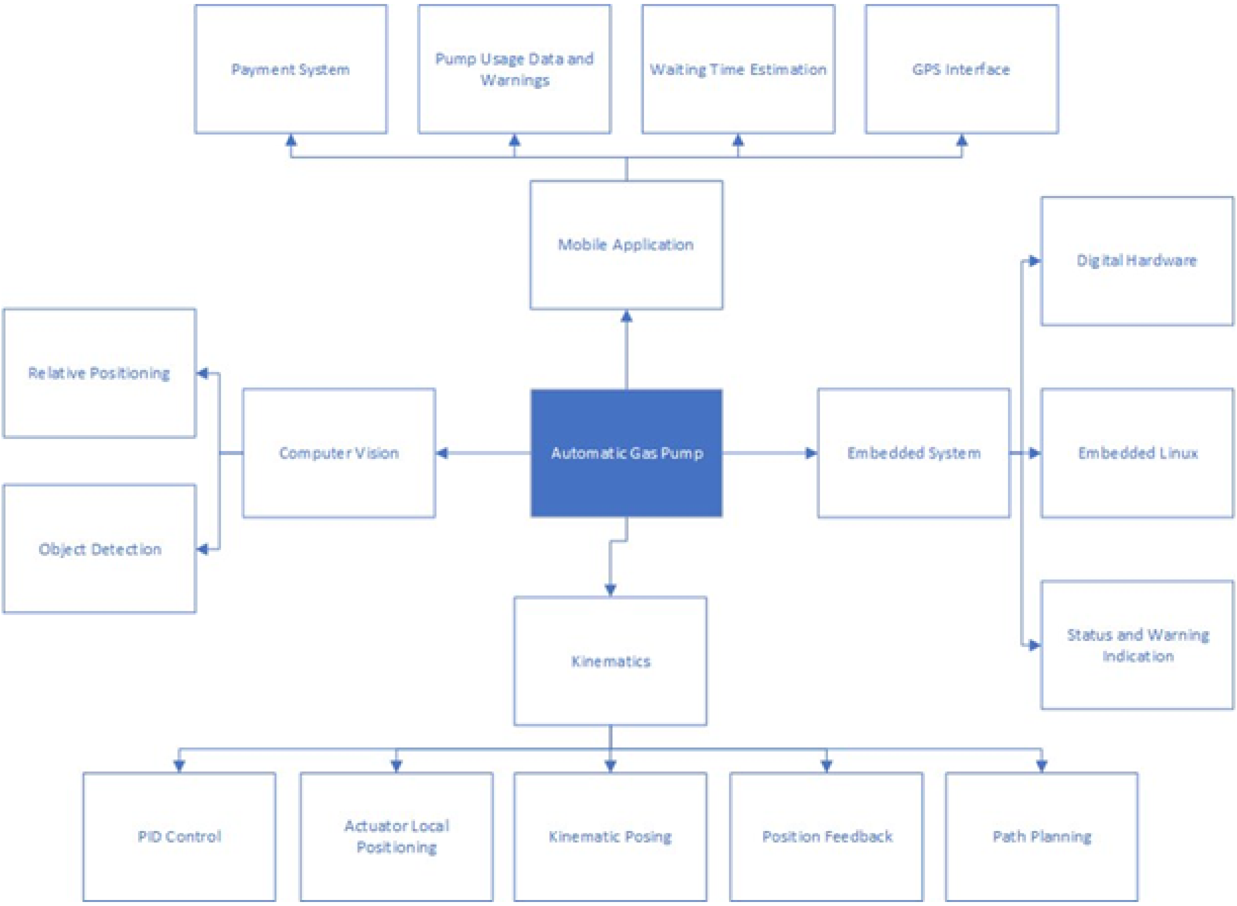


Fig. 30: System Level Overview

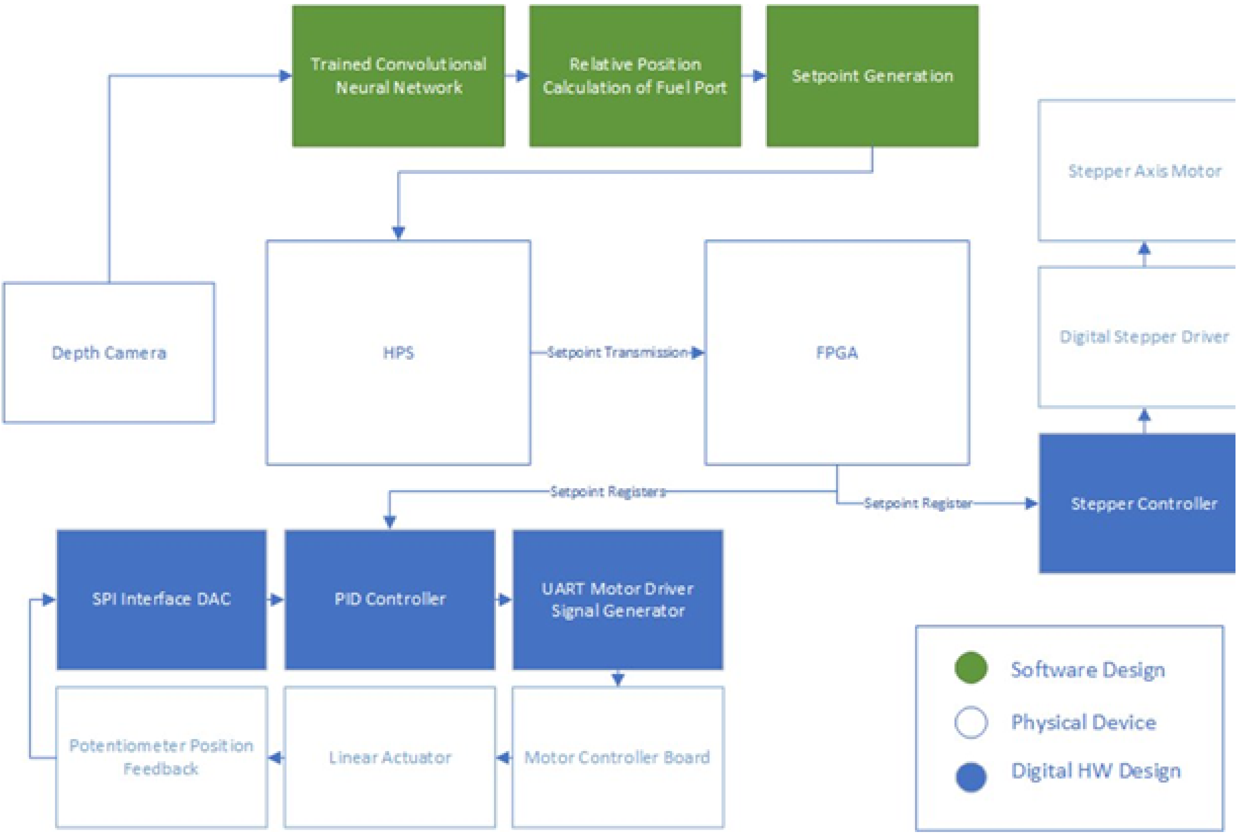


Fig. 31: Software Block Diagram

XVI. Appendix D: Mechanical Aspects

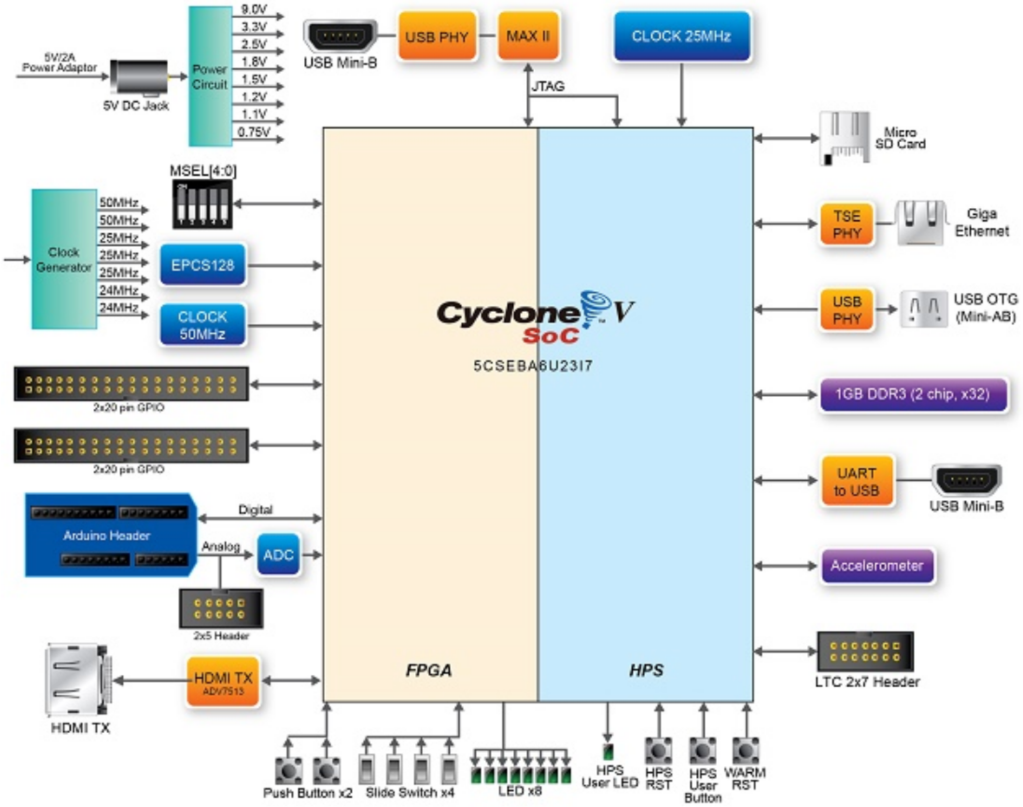


Fig. 32: DE-10 Nano Block Diagram

XVII. APPENDIX E: RESUMES

Michael Gomez

m.nathan.gomez@gmail.com • (707) 249-1659
Portfolio • Github • LinkedIn

Experience

SimpliSafe
Firmware Engineer II **March ‘21 - Present**

- Implementing reliable, high-performance firmware for low power home security products based on 16-bit and 32-bit microcontroller architecture (PIC16, ARM Cortex-M)
- Leading end-to-end firmware design for a novel home security product.
- Collaborating with vendors such as TI, Microchip, and ST to select parts appropriate to the project use-case.
- Assisting third party manufacturers when needed with debug firmware builds and board bring-up.
- Successfully proposed a partial Rust firmware re-write, spearheading Rust evangelism throughout the company

SimpliSafe
Firmware Engineer Co-op **January ‘21 - March ‘21**

Aruba, a Hewlett Packard Enterprise Company
Embedded Software Engineering Intern **May ‘20 – August ‘20**

- Developed 3 VxLAN feature tests in Python to check functionality in edge-case scenarios and detect regressions. Capitalized on Pytest fixtures to permute through potential network switch configurations.
- Consolidated network switch command line debugging tools by reducing required input complexity by a factor of 10.
- Participated in daily stand-up meetings, code reviews, and team demos. Collaborated closely with Infrastructure and DevOps teams to deliver high-quality product code.

Aruba, a Hewlett Packard Enterprise Company
Networking R&D Hardware Design Intern **May ‘19 – August ‘19**

- Enabled prototype board 'bring-up' immediately upon board delivery date by developing Python scripts to exercise key communications interfaces (I2C, SPI) and device functionality.
- Consulted data sheets to write low-level device drivers for new parts on the prototype.
- Deployed test scripts on network switch Linux-based OS.
- Analyzed switch signal integrity with oscilloscope and logic analyzer.

Skills

Languages: C, Rust, C++, Python, Verilog, HTML, CSS, Javascript
Frameworks: Flask, Django, Bulma, Tailwind, Gatsby, Node, Express
Software Tools: Jira, Git, Gerrit, VSCode, Matlab, GDB, PDB, Scapy, Pytest, vim, Linux, Make, Valgrind
Hardware Tools: Oscilloscope, function generator, logic analyzer, ARM DS-5, STM32Cube, MPLAB, JTAG

Education

California State University, Sacramento • GPA: 3.6
Bachelor’s of Science, Computer Engineering • May ‘21

- *President*, Association for Computing Machinery. Planned and led technical workshops (Intro to Python, Intro to Git).
- International Collegiate Programming Competition Regional Qualifier, Fall 2019

Steven Hellem

Current electrical and electronics engineering student, well-versed in mathematics, science, and physics with a keen interest in control systems.

Experience

Robotics
Student
August ‘20 – October ‘20

- Worked in a team environment to develop a differential drive robot.
- Trouble shooting sensors and other components to meet deadlines.
- Programming PWM values

Microcontroller Programming
Student
Jan ’19 – May ‘19

- Utilizing AD2 board, Raspberry Pi, Arduino, Parallax to complete assigned tasks.
- Wrote high level reports documenting work done.
- Created home automation system utilizing Raspberry Pi and numerous sensors.

Differential Drive Robot

August ’20 – October ‘20

- Designed a differential drive robot that completes course requirements.

Senior Design Project
August ‘20 – December ‘20

- Designed and reported on development of automatic gas pump.

Skills

Languages: MATLAB, Verilog
Software Tools: Git, Matlab, Quartus, PSpice
Hardware Tools: Oscilloscope, function generator

Education

California State University, Sacramento • GPA: 3.5
Bachelor’s of Science, Electrical and Electronics Engineering• May ‘21

Ahmed Abdel-Gwad

Experienced computer engineering student aiming to apply a varied skill set across embedded systems and firmware design domains.

Experience

Lawrence Livermore National Laboratory

Controls Engineering Intern May ‘20 – Present

- Performed a risk assessment of the motion control system of a machine used in the production of precision optics for the National Ignition Facility.
- Surveyed options for an updated motion control system to reduce risk and improve performance in an optics fabrication machine.
- Created block diagrams of the existing motion control system.
- Gathered requirements and developed a statement of work based on a Systems Engineering V model.

Sacramento State University

Student Assistant October‘17 – Present

- Assisted the electrical engineering department technician in lab maintenance, test equipment troubleshooting and repair, as well as assisting faculty with technical assistance.
- Assisted labs in courses related to digital logic design and robotics through in class tutoring.
- Developed a rapid prototyping strategy for antenna fabrication using a CNC router. Students in the antenna design laboratory designed dipole or microstrip antenna which were able to be quickly fabricated using different substrates and accuracy of CNC machining.
- Taught students soldering techniques for wire, SMD, and thru-hole soldering.

Robotic Manipulator November‘18 – December ‘18

- Designed a robotic arm with 3 degrees of freedom with an implementation of inverse kinematics. Kinematics were modeled in Matlab, then implemented in C++ on a microcontroller controlling servo motors at each joint.

Lidar Data Collection March ‘19 – April ‘19

- Interfaced with a Quanergy M8 Lidar using Python to generate a TCP stream. The packet binary data was then unpacked and written to a spreadsheet file for future use in Matlab. Minimal documentation was originally available except for a description of the packet structure transmitted by the device. Wireshark was used in combination with a visualization utility to confirm the packet structure before developing a data collection script in Python.

Parallax Quad Rover Upgrade April ‘18 - May ‘18

- Redesigned a Parallax quad rover from a gasoline/hydraulic drive to electric/hydraulic drive. Relays in combination with solenoids were used to control the flow of hydraulic fluid which controlled the speed and direction of movement on the rover.
- Design a system for control of the rover over a network connection for use over long distance.

Skills

Languages: C, C++, Python, Verilog, Java
Software Tools: Git, VSCode, Matlab, Autodesk Fusion 360, Adobe Suite, PSpice
Hardware Tools: Oscilloscope, function generator, logic analyzer, Virtual Test Bench, Soldering

Education

California State University, Sacramento • GPA: 3.1
Bachelor’s of Science, Computer Engineering • May ‘21

Jordan Moore

A current electrical and electronic engineering student aiming to apply a varied skill set in the field of control systems and robotics.

Experience

3QC Inc.
Engineering Intern May ‘19 – August ‘19

- Analyzed technical documents and construction blueprints for information extrapolation
- Conducted site walks for on-site testing for different construction projects
- Working a team environment consisting of both engineers and non-engineers

Robotics
Student August‘20 –October‘20

- Worked alone to design and build a differential drive robot
- Trouble shooting sensors and other components to meet deadlines.
- Calculated and programed PWM values

Hyperloop Pod June ‘17 – August ‘17

- Worked in a team to construct a Hyperloop pod for SpaceX’s Hyperloop competition
- Wired and connected electrical and controls systems to the pod
- Worked with the mechanical team to implement the electrical controls and sensors with the mechanical aspects of the pod

Fire Detection System January ‘20 – May ‘20

- Used OpenCv, a computer vision library, to create a trainable machine vision system
- Integrated OpenCv with Rasberry pi to allow C code to interface with thermal camera
- Trained machine vision system using depth learning functions to isolate specific characteristics of a still image or video to detect the presence of fire

Differential Drive Robot August ‘20 –October‘20

- Designed a 4 wheeled differential drive robot to complete assignment-based tasks

Skills

Languages: C, C++, Python, HTML, CSS
Software Tools: Git, VSCode, Matlab, Scapy, Pytest
Hardware Tools: Oscilloscope, function generator, logic analyzer, Quartus, STM32Cube

Education

California State University, Sacramento • GPA: 3.1
Bachelor’s of Science, Electrical and Electronic Engineering • May ‘21

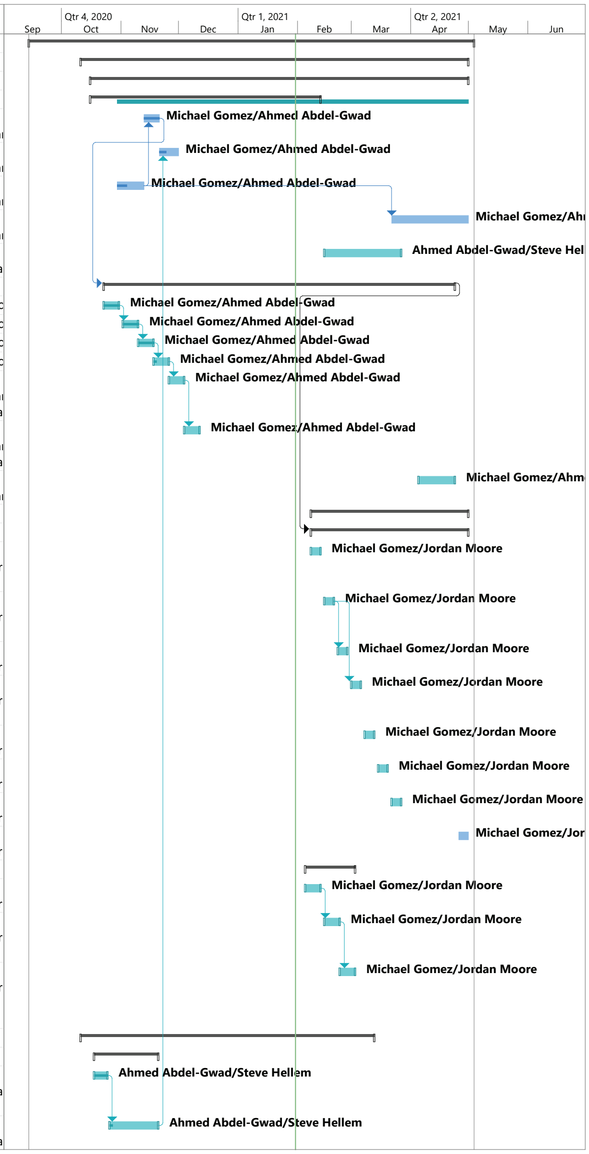
XVIII. APPENDIX F: TABULAR WBS

Robogas Work Breakdown Structure			Responsible Staff	Scope
1. Embedded System	1.1 Digital Hardware	1.1.1 Load "golden hardware reference design" to FPGA	Michael Gomez/Ahmed Abdel-Gwad	10/16 - 10/23
		1.1.2 Instantiate PID modules for motor control	Michael Gomez/Ahmed Abdel-Gwad	10/30 - 11/6
		1.1.3 Use Qsys to define HPS-FPGA interface	Michael Gomez/Ahmed Abdel-Gwad	10/30 - 11/6
		1.1.4 Implement hardware accelerated image processing	Michael Gomez/Ahmed Abdel-Gwad	3/22/21 - 4/30/21
	1.2 Embedded Linux	1.2.1 Generate preloader	Michael Gomez/Ahmed Abdel-Gwad	10/23 - 10/30
		1.2.2 Genereate bootloader	Michael Gomez/Ahmed Abdel-Gwad	10/23 - 10/30
		1.2.3 Generate device tree	Michael Gomez/Ahmed Abdel-Gwad	10/23 - 10/30
		1.2.4 Build kernel	Michael Gomez/Ahmed Abdel-Gwad	10/23 - 10/30
		1.2.5 Develop applications to run on custom kernel, and interact with FPGA IP	Michael Gomez/Ahmed Abdel-Gwad	11/6 - 11/13
		1.2.6 Develop Linux application built on Intel Realsense SDK 2.0 to process image data and drive custom motor controls	Michael Gomez/Ahmed Abdel-Gwad	11/13 - 11/20
		1.2.7 Develop web server to interact with mobile application	Michael Gomez/Jordan Moore	4/5/21 - 4/23-21
2. Mobile Application	2.1 Back-end	2.1.1 Signal robot arm to begin movement program after succesful payment	Michael Gomez/Jordan Moore	2/8/21 - 2/12/21
		2.1.2 Create MySQL database to store user information	Michael Gomez/Jordan Moore	2/15/21 - 2/19/21
		2.1.3 Create a REST API using Flask to interact with database	Michael Gomez/Jordan Moore	2/22/21 - 2/26/21
		2.1.4 Report streaming pump usage data to user	Michael Gomez/Jordan Moore	3/1/21 - 3/5/21
		2.1.5 Report malfunctioning pump to user	Michael Gomez/Jordan Moore	3/8/21 - 3/12/21
		2.1.6 Integrate Google Maps API for easy directions to nearest station	Michael Gomez/Jordan Moore	3/15/21 - 3/19/21
		2.1.7 Integrate Flask back end with React Native front end	Michael Gomez/Jordan Moore	3/22/21 - 3/26/21
		2.1.8 Integrate Stripe API for secure payments	Michael Gomez/Jordan Moore	4/26/21 - 4/30/21
	2.2 Front-End	2.6.1 Create a logo/banner	Michael Gomez/Jordan Moore	2/5/21 - 2/12/21
		2.6.2 Create a wireframe mockup of the application	Michael Gomez/Jordan Moore	2/15/21 - 2/22/21
		2.6.3 Use React Native to create a cross-platform front-end	Michael Gomez/Jordan Moore	2/23/21 - 3/2/21
3. Kinematics	3.1 PID Control	3.1.1 Tuning and Gain value identification and optimization test (arduino)	Ahmed Abdel-Gwad/Steve Hellem	10/18 - 10/24
		3.1.2 Implement PID Controller on FPGA	Ahmed Abdel-Gwad/Steve Hellem	10/25 - 10/31
			Ahmed Abdel-Gwad/Steve Hellem	10/31 - 11/1
	3.2 Actuator Local Positioning	3.2.1 Creat function to translate coordinates to position values for each actuator	Ahmed Abdel-Gwad/Steve Hellem	11/1 - 11/7
		3.2.2 Module to keep track of positions in system	Ahmed Abdel-Gwad/Steve Hellem	11/6 - 11/8
		3.2.3 Create Stepper Motor homing/position tracker	Ahmed Abdel-Gwad/Steve Hellem	11/8 - 11/21
		3.2.4 Function to handle out of bounds position requests	Ahmed Abdel-Gwad/Steve Hellem	11/14 - 11/15
	3.3 Kinematic Posing	3.3.1 Create inverse kinematics function (generate actuator positioning from 3D coordinate input)	Ahmed Abdel-Gwad/Steve Hellem	11/1 - 11/7
		3.3.2 Function to pose at idle position	Ahmed Abdel-Gwad/Steve Hellem	11/7 - 11/8
		3.3.3 Function to pose in front of gas tank from coordinate generated from controller	Ahmed Abdel-Gwad/Steve Hellem	11/7 - 11/8
	3.4 Position Feedback	3.4.1 Calibrate Linear Actuator Feedback Devices	Ahmed Abdel-Gwad/Steve Hellem	10/11 - 10/17
		3.4.2 Analyze motor feedback signal and generate filter if needed	Ahmed Abdel-Gwad/Steve Hellem	10/25 - 10/31
		3.4.3 Evaluate range of travel signals and repeatability	Ahmed Abdel-Gwad/Steve Hellem	10/25 - 10/31
	3.5 Path Planning	3.5.1 Function to generate path to guide nozzle into gas tank	Ahmed Abdel-Gwad/Steve Hellem	11/22 - 11/28
		3.5.2 Function to generate shortest path to 3D coordinate	Ahmed Abdel-Gwad/Steve Hellem	11/29 - 12/5
4. Computer vision	4.1 Object Detection	4.1.1 Create test module for object detection using YOLOv3	Jordan Moore	10/16 - 10/23
		4.1.2 Use OpenCV to implement test module	Jordan Moore	10/23 - 10/30
		4.1.3 Calibrate OpenCV system	Jordan Moore	10/30 - 11/6
		4.1.4 Collect photographs of gashole	Jordan Moore	10/30 - 11/6
		4.1.5 Annotate gashole photographs for training	Jordan Moore	11/6 - 11/13
		4.1.6 Train custom YOLOv3 module with gashole pictures	Jordan Moore	11/6 - 11/13
		4.1.7 Use OpenCV to test trained object detection module	Jordan Moore	11/13 - 11/20
		4.1.8 Calibrate the bounding boxes on detected object	Jordan Moore	11/13 - 11/20
	4.2 Relative Position	4.2.1 Intergrate Object Detection with Matlab	Jordan Moore	11/20 - 11/27
		4.2.2 Use bounding boxes on detected object to find center of gas hole	Jordan Moore	11/20 - 11/27
		4.2.3 Report bounding box and gashole center to control through Matlab	Jordan Moore /Steve Hellem	11/27 - 12/4
		4.2.4 Intergrate depth data with object detection	Jordan Moore /Steve Hellem	11/27 - 12/4
		4.2.5 Translate computer vision data to kinematic positional data	Jordan Moore /Steve Hellem	11/27 - 12/4
		4.2.6 Calibrate camera placement	Jordan Moore/Ahmed Abdel-Gwad	12/4 - 12/11
		4.2.7 Calibrate kinematic feedback based on machine vision data	Jordan Moore /Steve Hellem	12/4 - 12/11

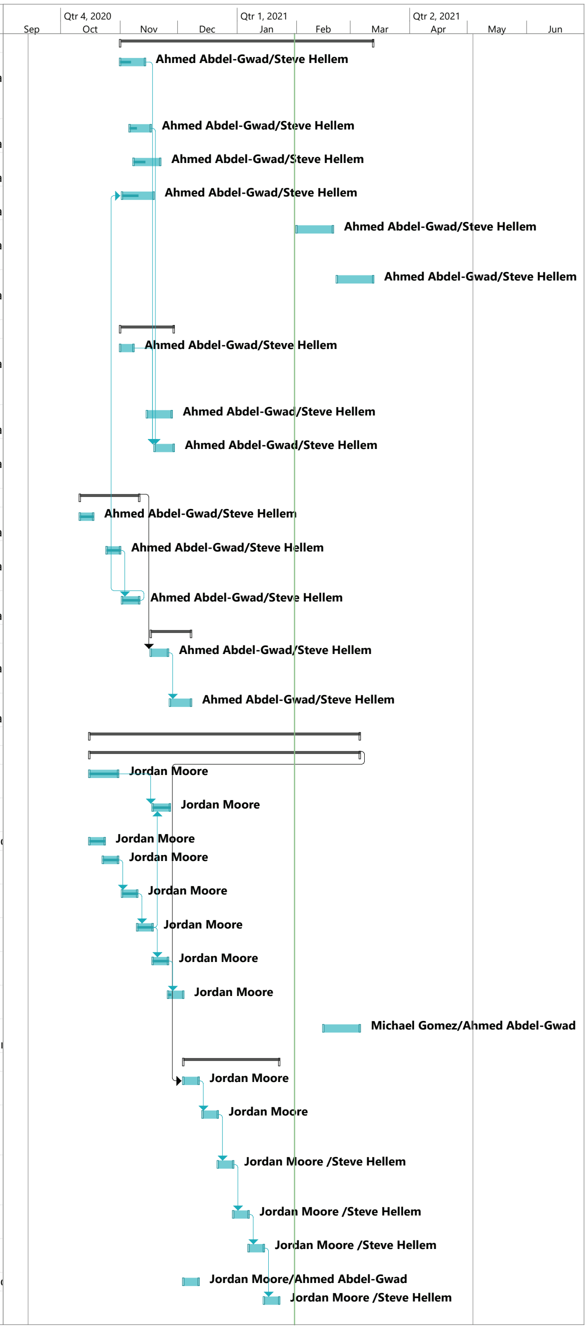
Fig. 33: Tabular WBS

XIX. Appendix G: Project Timeline

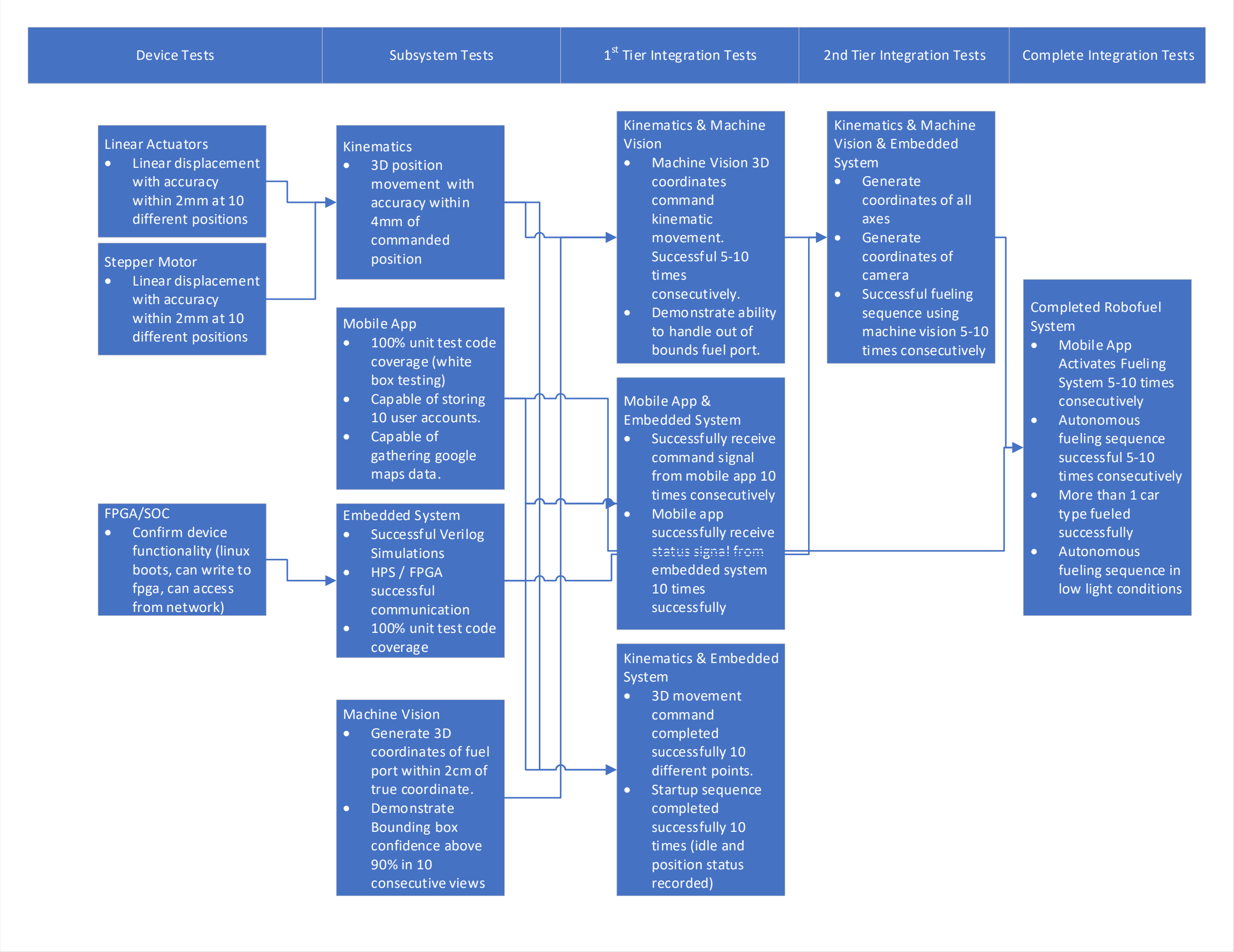
ID	Task Name	Duration	Start	Finish	Predecessors	Resource Names
1	**Senior Design**	**166 days?**	**Mon 9/14/2(**	**Mon 5/3/21**		
2	**Project Features**	**145 days?**	**Sun 10/11/2(**	**Fri 4/30/21**		
3	**1. Embedded System**	**141 days?**	**Fri 10/16/20**	**Fri 4/30/21**		
4	**1.1 Digital Hardware**	**86 days?**	**Fri 10/16/20**	**Fri 2/12/21**		
5	1.1.1 Load "golden hardware reference design" to FPGA	6 days	Fri 11/13/20	Fri 11/20/20	7	Michael Gomez/Ah
6	1.1.2 Instantiate PID modules for motor control	6 days	Sat 11/21/20	Mon 11/30/20	35	Michael Gomez/Ah
7	1.1.3 Use Qsys to define HPS-FPGA interface	10 days	Fri 10/30/20	Thu 11/12/20		Michael Gomez/Ah
8	1.1.4 Implement hardware accelerated image processing	30 days	Mon 3/22/21	Fri 4/30/21	7	Michael Gomez/Ah
9	1.1.5 Increase HW/SW Controls Stability	30 days	Mon 2/15/21	Fri 3/26/21		Ahmed Abdel-Gwa
10	**1.2 Embedded Linux**	**131 days**	**Fri 10/23/20**	**Fri 4/23/21**	**5**	
11	1.2.1 Generate preloader	6 days	Fri 10/23/20	Fri 10/30/20		Michael Gc
12	1.2.2 Genereate bootloader	6 days	Mon 11/2/2(	Mon 11/9/20	11	Michael Gc
13	1.2.3 Generate device tree	6 days	Tue 11/10/2(	Tue 11/17/20	12	Michael Gc
14	1.2.4 Build kernel	6 days	Wed 11/18/2	Wed 11/25/20	13	Michael Gc
15	1.2.5 Develop applications to run on custom kernel, and interact with FPGA IP	6 days	Thu 11/26/20	Thu 12/3/20	14	Michael Gomez/Ah Abdel-Gwa
16	1.2.6 Develop Linux application built on Intel Realsense SDK 2.0 to process image	6 days	Fri 12/4/20	Fri 12/11/20	15	Michael Gomez/Ah Abdel-Gwa
17	1.2.7 Develop Server to Interface With Mobile App	15 days	Mon 4/5/21	Fri 4/23/21		Michael Gomez/Ah
18	**2. Mobile Application**	**60 days**	**Mon 2/8/21**	**Fri 4/30/21**		
19	**2.1 Back-end**	**60 days**	**Mon 2/8/21**	**Fri 4/30/21**	**10**	
20	2.1.1 Signal robot arm to begin movement program after succesful payment	5 days	Mon 2/8/21	Fri 2/12/21		Michael Gomez/Jor Moore
21	2.1.2 Create a REST API using Flask to interact with databaseTask	5 days	Mon 2/15/21	Fri 2/19/21		Michael Gomez/Jor Moore
22	2.1.3 Create MySQL database to store user information	5 days	Mon 2/22/21	Fri 2/26/21	21	Michael Gomez/Jor
23	2.1.4 Integrate Google Maps API for easy directions to nearest station	5 days	Mon 3/1/21	Fri 3/5/21	21	Michael Gomez/Jor Moore
24	2.1.5 Integrate Flask back end with React Native front end	5 days	Mon 3/8/21	Fri 3/12/21		Michael Gomez/Jor
25	2.1.6 Report streaming pump usage data to user	5 days	Mon 3/15/21	Fri 3/19/21		Michael Gomez/Jor
26	2.1.7 Report malfunctioning pump to user	5 days	Mon 3/22/21	Fri 3/26/21		Michael Gomez/Jor
27	2.1.8 Integrate Stripe API for secure payments	5 days	Mon 4/26/21	Fri 4/30/21		Michael Gomez/Jor
28	**2.2 Front-End**	**18 days**	**Fri 2/5/21**	**Tue 3/2/21**		
29	2.2.1 Create a logo/banner	6 days	Fri 2/5/21	Fri 2/12/21		Michael Gomez/Jor
30	2.2.2 Create a wireframe mockup of the application	6 days	Mon 2/15/21	Mon 2/22/21	29	Michael Gomez/Jor Moore
31	2.2.3 Use React Native to create a cross-platform front-end	6 days	Tue 2/23/21	Tue 3/2/21	30	Michael Gomez/Jor Moore
32	**3. Kinematics**	**110 days**	**Sun 10/11/2(**	**Fri 3/12/21**		
33	**3.1 PID Control**	**24.5 days**	**Sun 10/18/2(**	**Fri 11/20/20**		
34	3.1.1 Tuning and Gain value identification and optimization test (arduino)	7 days	Sun 10/18/20	Sat 10/24/20		Ahmed Abdel-Gwa Hellem
35	3.1.2 Implement PID Controller on FPGA	19.5 days	Mon 10/26/20	Fri 11/20/20	34	Ahmed Abdel-Gwa



ID	Task Name	Duration	Start	Finish	Predecessors	Resource Names
36	**3.2 Actuator Local Positioning**	**95 days**	**Sun 11/1/20**	**Fri 3/12/21**		
37	3.2.1 Creat function to translate coordinates to position values for each actuator	11 days	Sun 11/1/20	Fri 11/13/20		Ahmed Abdel-Gwa Hellem
38	3.2.2 Module to keep track of positions in system	7 days	Fri 11/6/20	Mon 11/16/20		Ahmed Abdel-Gwa
39	3.2.3 Create Stepper Motor homing/position tracker	12 days	Sun 11/8/20	Sat 11/21/20		Ahmed Abdel-Gwa
40	3.2.4 Function to handle out of bounds position requests	12.5 days	Mon 11/2/20	Wed 11/18/20	50	Ahmed Abdel-Gwa
41	3.2.5 Design and Deploy Frame Improvements for Accuracy	15 days	Mon 2/1/21	Fri 2/19/21		Ahmed Abdel-Gwa Hellem
42	3.2.6 Swap Feedback Hardware For Quadrature Encoders	15 days	Mon 2/22/21	Fri 3/12/21		Ahmed Abdel-Gwa Hellem
43	**3.3 Kinematic Posing**	**20 days**	**Sun 11/1/20**	**Sat 11/28/20**		
44	3.3.1 Create inverse kinematics function (generate actuator positioning from 3D coordinate input)	7 days	Sun 11/1/20	Sat 11/7/20		Ahmed Abdel-Gwa Hellem
45	3.3.2 Function to pose at idle position	11 days	Sun 11/15/20	Fri 11/27/20		Ahmed Abdel-Gwa
46	3.3.3 Function to pose in front of gas tank from coordinate generated from controller	8 days	Thu 11/19/20	Sat 11/28/20	37,38,44	Ahmed Abdel-Gwa Hellem
47	**3.4 Position Feedback**	**22 days**	**Sun 10/11/2(**	**Tue 11/10/20**		
48	3.4.1 Calibrate Linear Actuator Feedback Devices	7 days	Sun 10/11/20	Sat 10/17/20		Ahmed Abdel-Gwa
49	3.4.2 Analyze motor feedback signal and generate filter if needed	7 days	Sun 10/25/20	Sat 10/31/20		Ahmed Abdel-Gwa Hellem
50	3.4.3 Evaluate range of travel signals and repeatability	7 days	Mon 11/2/20	Tue 11/10/20	49	Ahmed Abdel-Gwa
51	**3.5 Path Planning**	**15 days**	**Tue 11/17/2(**	**Mon 12/7/20**		
52	3.5.1 Function to generate path to guide nozzle into gas tank	7 days	Tue 11/17/20	Wed 11/25/20	47	Ahmed Abdel-Gwa Hellem
53	3.5.2 Function to generate shortest path to 3D coordinate	7 days	Fri 11/27/20	Mon 12/7/20	52	Ahmed Abdel-Gwa
54	**4. Computer vision**	**101 days**	**Fri 10/16/20**	**Fri 3/5/21**		
55	**4.1 Object Detection**	**101 days**	**Fri 10/16/20**	**Fri 3/5/21**		
56	4.1.1 Create test module for object detection using YOLOv3	11 days	Fri 10/16/20	Fri 10/30/20		Jordan Moore
57	4.1.2 Use OpenCV to implement test module	7 days	Wed 11/18/20	Thu 11/26/20	56,61	Jordan Moore
58	4.1.3 Calibrate OpenCV system	6 days	Fri 10/16/20	Fri 10/23/20		Jordan Mo
59	4.1.4 Collect photographs of gashole	6 days	Fri 10/23/20	Fri 10/30/20		Jordan Moore
60	4.1.5 Annotate gashole photographs for training	6 days	Mon 11/2/20	Mon 11/9/20	59	Jordan Moore
61	4.1.6 Train custom YOLOv3 module with gashole pictures	6 days	Tue 11/10/20	Tue 11/17/20	60	Jordan Moore
62	4.1.7 Use OpenCV to test trained object detection module	6 days	Wed 11/18/20	Wed 11/25/20	61	Jordan Moore
63	4.1.8 Calibrate the bounding boxes on detected object	6 days	Thu 11/26/20	Thu 12/3/20	62	Jordan Moore
64	4.1.9 Calibrate and Refine 3D Coordinate Generation	15 days	Mon 2/15/21	Fri 3/5/21		Michael Gomez/Ahı
65	**4.2 Relative Position**	**36 days**	**Fri 12/4/20**	**Fri 1/22/21**		
66	4.2.1 Intergrate Object Detection with Matlab	6 days	Fri 12/4/20	Fri 12/11/20	55	Jordan Moore
67	4.2.2 Use bounding boxes on detected object to find center of gas hole	6 days	Mon 12/14/20	Mon 12/21/20	66	Jordan Moore
68	4.2.3 Report bounding box and gashole center to control through Matlab	6 days	Tue 12/22/20	Tue 12/29/20	67	Jordan Moore /Steve
69	4.2.4 Intergrate depth data with object detection	6 days	Wed 12/30/20	Wed 1/6/21	68	Jordan Moore
70	4.2.5 Translate computer vision data to kinematic positional data	6 days	Thu 1/7/21	Thu 1/14/21	69	Jordan Moore
71	4.2.6 Calibrate camera placement	6 days	Fri 12/4/20	Fri 12/11/20		Jordan Mo
72	4.2.7 Calibrate kinematic feedback based on machine	6 days	Fri 1/15/21	Fri 1/22/21	70	Jordan Moore



XX. APPENDIX H: DEVICE TEST PLAN



Tasks	**Device Level**		Date Range	**Team members**
	Linear Actuators			
		Linear displacement with accuracy within 2 mm at 10 different positions	2/22 - 2/26	Ahmed Abdel-Gwad
	Stepper Motors			
		Linear displacement with accuracy within 2 mm at 10 different positions	2/22 - 2/26	Steve Hellem
	FPGA/SOC			
		Confirm device functionality - Booting into linux, writing to FPGA, network access	2/22 - 2/26	Michael Gomez
	Subsystem Tests			
	Mobile App			
		100% unit test code coverage (white box testing)	3/29 - 4/2	Michael Gomez
		Capable of storing 10 user accounts	3/29 - 4/2	Jordan Moore
		Capable of gathering Google Maps data	3/22 - 3/26	Jordan Moore
	Embedded System			
		Succesful Verilog simulations (whitebox)	3/1 - 3/5	Ahmed Abdel-Gwad
		Succesful HPS/FPGA interfacing	3/1 - 3/5	Michael Gomez
		100% unit test code coverage (white box testing)	3/8 - 3/12	Michael Gomez
	Machine Vision			
		Generate 3D coordinates of fuel ports within 2cm of true coordinate	3/15 - 3/19	Michael Gomez/Jordan Moore
		Demonstrate bounding box confidence above 90% in 10 consecutive attempts	3/22 - 3/26	Jordan Moore
	Kinematics			
		3D position movement with accuracy within 4mm of commanded position	3/8 - 3/12	Steve Hellem
	1st Tier Integration Tests			
	Kinematics & Machine Vision			
		Machine Vision 3D coordinates command kinematic movement. Succesful 5-10 times.	3/22 - 3/26	Ahmed Abdel-Gwad
		Demonstrate ability to handle out of bounds fuel port	3/29 - 4/2	Ahmed Abdel-Gwad
	Mobile App & Embedded System			
		Succesfully receive command signal from mobile app ten times consecutively	3/22 - 3/26	Jordan Moore
		Mobile app succesfully receives status signal from embedded system ten times.	3/22 - 3/26	Jordan Moore
	Kinematics & Embedded System			
		3D movement command completed succesfully at 10 different points.	3/22 - 3/26	Steve Hellem
		Startup sequence completed succesfully 10 times (idle and position status recorded)	3/22 - 3/26	Steve Hellem
	2nd Tier Integration Tests			
	Kinematics & Machine Vision & Embedded system	Generate coordiantes of all axes	3/29 - 4/2	Michael Gomez
		Generate coordinates of camera	3/29 - 4/2	Ahmed Abdel-Gwad
		Succesful fueling sequence using machine vision 5-10 times consecutively	3/29 - 4/2	Steve Hellem
	Complete integration tests			
		Mobile app activates fueling system 5-10 times consecutively	3/29 - 4/2	Jordan Moore
		Autonomous fueling sequence succesful 5-10 times consecutively	3/29 - 4/2	Michael Gomez
		More than 1 car type fueled succesfully	3/29 - 4/2	Ahmed Abdel-Gwad
		Autonomous fueling sequence in low light conditions	3/29 - 4/2	Steve Hellem

Tasks	Device Level		Date Range	Team members
1a	**Linear Actuators**	🟩 Linear displacement with accuracy within 2 mm at 10 different positions	2/22 - 2/26	Ahmed Abdel-Gwad
1b	Stepper Motors	🟩 Linear displacement with accuracy within 2 mm at 10 different positions	2/22 - 2/26	Steve Hellem
1c	FPGA/SOC		2/22 - 2/26	Michael Gomez
2	**Subsystem Tests**			
2a	**Mobile App**	🟩 100% unit test code coverage (white box testing)	3/29 - 4/2	Michael Gomez
		🟩 Capable of storing 10 user accounts	3/29 - 4/2	Jordan Moore
		🟩 Capable of gathering Google Maps data	3/22 - 3/26	Jordan Moore
2b	**Embedded System**	🟩 Succesful Verilog simulations (whitebox)	3/1 - 3/5	Ahmed Abdel-Gwad
		🟩 Succesful HPS/FPGA interfacing	3/1 - 3/5	Michael Gomez
		🟩 100% unit test code coverage (white box testing)	3/8 - 3/12	Michael Gomez
2c	**Machine Vision**	🟩 Generate 3D coordinates of fuel ports within 2cm of true coordinate	3/15 - 3/19	Michael Gomez/Jordan Moore
		🟨 Demonstrate bounding box confidence above 90% in 10 consecutive attempts	3/22 - 3/26	Jordan Moore
2d	Kinematics	🟩 3D position movement with accuracy within 4mm of commanded position	3/8 - 3/12	Steve Hellem
3	**1st Tier Integration Tests**			
3a	Kinematics & Machine Vision	🟩 Machine Vision 3D coordinates command kinematic movement. Succesful 5-10 times.	3/22 - 3/26	Ahmed Abdel-Gwad
		🟨 Demonstrate ability to handle out of bounds fuel port	3/29 - 4/2	Ahmed Abdel-Gwad
3b	**Mobile App & Embedded System**	🟨 Succesfully receive command signal from mobile app ten times consecutively	3/22 - 3/26	Jordan Moore
		🟨 Mobile app succesfully receives status signal from embedded system ten times.	3/22 - 3/26	Jordan Moore
3c	Kinematics & Embedded System	🟩 **3D movement command completed succesfully at 10 different points.**	3/22 - 3/26	Steve Hellem
		🟩 Startup sequence completed succesfully 10 times (idle and position status recorded)	3/22 - 3/26	Steve Hellem
4	**2nd Tier Integration Tests**			
4a	**Kinematics & Machine Vision & Embedded system**	🟩 Generate coordiantes of all axes	3/29 - 4/2	Michael Gomez
		🟩 Generate coordinates of camera	3/29 - 4/2	Ahmed Abdel-Gwad
		🟩 Succesful fueling sequence using machine vision 5-10 times consecutively	3/29 - 4/2	Steve Hellem
4b	Complete integration tests	🟨 Mobile app activates fueling system 5-10 times consecutively	3/29 - 4/2	Jordan Moore
		🟨 Autonomous fueling sequence succesful 5-10 times consecutively	3/29 - 4/2	Michael Gomez
		🟨 More than 1 car type fueled successfully	3/29 - 4/2	Ahmed Abdel-Gwad
		🟨 Autonomous fueling sequence in low light conditions	3/29 - 4/2	Steve Hellem

🟩 Exceeds Exceptations

🟨 Meets Exceptations