SMART STAIR GATE

By

Alexander Chin

Brandon Lau

Zeyad Irsheid

Design Document for ECE 445, Senior Design, Fall 2023

TA: Zicheng Ma

28 September 2023

Project No. 37

Contents

1 Introduction

1.1 Problem

In today's society, parents have more things than ever to worry about especially parents of crawling children. Children are unpredictable at times especially when unsupervised, and the last thing parents want is their child to be doing something dangerous. One of the most dangerous activities a baby can do unsupervised is crawl down a flight of stairs. According to researchers, "Every six minutes, a child falls down stairs somewhere in the U.S., experts say. Over 90,000 kids under the age of 5 end up in emergency rooms because of stair-related falls every year" **[1]**. Although manual gates exist to solve this issue, parents may still forget to close the gate behind them leaving children susceptible to potentially life-threatening accidents.

1.2 Solution

Our solution provides peace of mind for parents of crawling children. We propose to create a smart gate enclosed at the top of a flight of stairs. The smart gate will be initially closed, and open if an adult or an animal approaches the gate.

This solution contains two main sub-systems: detecting when a baby approaches the gate and preventing the baby from crawling down a flight of stairs. Our detection system will be integrated into the gate containing a PIR sensor, a proximity sensor, and RF sensors to determine whether or not a crawling baby is approaching the gate. Our prevention subsystem will contain a motorized gate that will prohibit a crawling baby from crawling down a flight of stairs. We also plan to include a wireless system that enables parents to remotely monitor and control the gate's status, allowing them to open and close the gate.

1.2.1 Visual Aid

Figure 1: Simple Visual Aid of the Smart Stair Gate

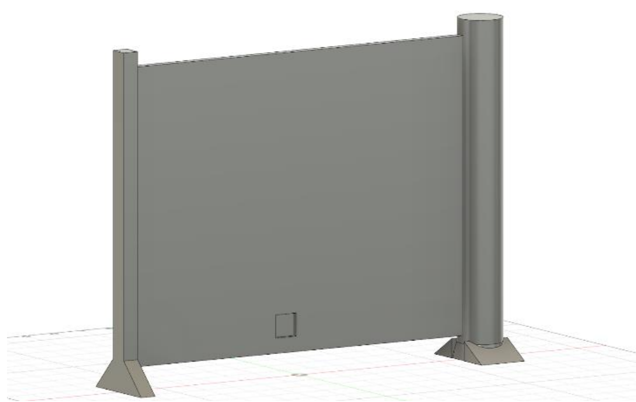


Figure 2: Front-side with sensor placement of the 18-inch wide gate

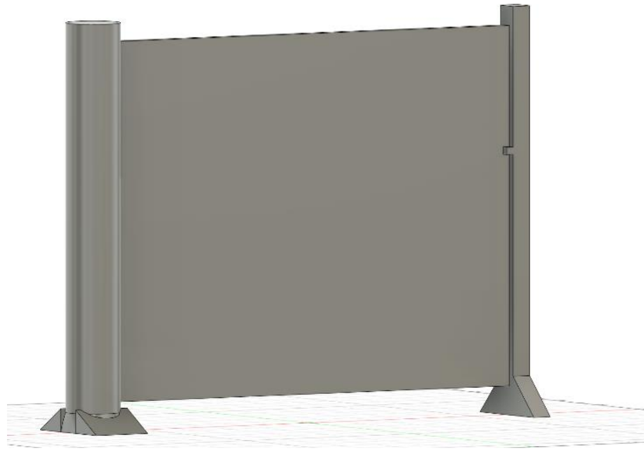


Figure 3: Back-side with solenoid placement

1.3 High-level Requirements

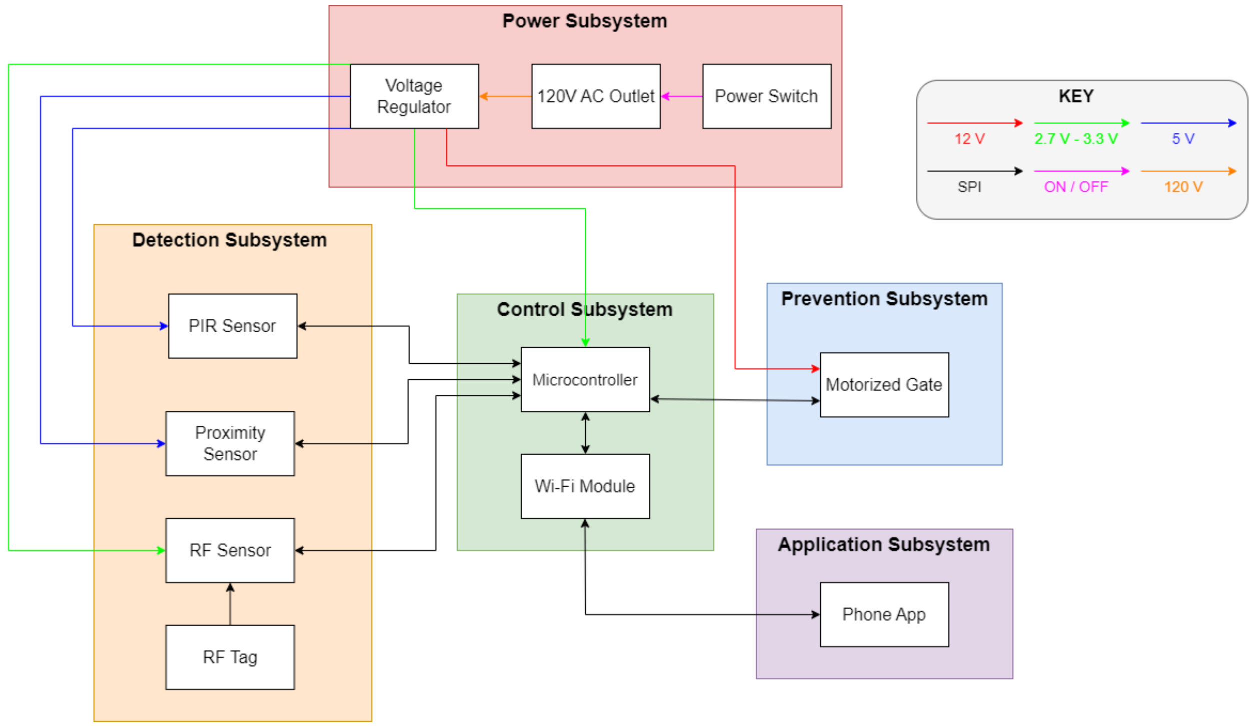
- Gate detection occurs when an object moves within a 3 ±. 5 foot proximity of the gate and sends a signal to the microcontroller.
    - The gate detection module will utilize information captured by a motion sensor. If the motion sensor is triggered, it will signal the microcontroller.

- Gate should be able to differentiate between a toddler, dog, and adult. The gate should automatically open if a dog or an adult wants to pass.
    - RFID tag should be detected within a 3 ± .5 foot radius. A toddler is assumed to have RFID on the body.
    - Gate should open when a dog or adult is within a 3 ± .25 foot radius. Should be able to tell if it is a human and not an object.

- Collision prevention should be active while the gate is in motion.
    - While the gate is in the process of opening/closing, there will be motion sensors active to detect if anything is near the gate. If any object is within 3 ± .5 inches of the gate door's front, it will stop until the object is no longer there.
    - After the object is gone, the gate will resume the former action

2 Design

2.1 Block Diagram



2.2 Subsystem Overview

Our product calls for five subsystems: power, control, detection, prevention, application.

2.2.1 Power Subsystem

The power system will connect to the power line and stepped down by a transformer. The AC signal will be fed into a rectifier turning it into a DC signal. With the use of filters, LDOs, and buck converters, we will get the voltages we need to safely power the microcontroller and the motors. The power line is used in the case that batteries run out of charge. An LED will be used to indicate the device's functionality. This subsystem will provide functionality and power to the control, detection, and prevention subsystem.

2.2.2 Control Subsystem

The control system is where all the signals are sent to control the product. This subsystem consists of our main PCB equipped with a microcontroller that will send and receive electrical signals. This subsystem will provide functionality to the detection, prevention, and application system as it will be, as the name suggests, control the behavior of all the parts.

2.2.3 Detection Subsystem

The detection system will allow the gate to tell if it should remain open or closed. It consists of multiple sensors including PIR, proximity, and RF sensors. The PIR allows the gate to differentiate between objects and living things; proximity sensors for when the device should be on standby and if anything is nearby when opening/closing; and RF sensors/tags to distinguish between toddlers and dogs. The PIR sensor will be always on while all the other sensors will be on standby until PIR sensor is triggered. This subsystem will provide information directly to the control subsystem and determines when the gate opens/closes (prevention subsystem).

2.2.4 Prevention Subsystem

The prevention system is everything that controls the mechanical parts of the products. This subsystem consists of a DC brushed gear motor w/encoder, the gate itself, and a solenoid lock. This subsystem's behavior is dictated by the control subsystem, but both the detection and application subsystem indirectly handles when the gate opens/closes.

2.2.5 Application Subsystem

The application subsystem allows for phone interaction. It consists of a Wi-Fi signal and a phone app. This phone app will be able to open and close the door from any location and will tell the user the state of the door. The application subsystem communicates with the control subsystem to determine and control the state of the gate (prevention subsystem). This subsystem will also override the detection subsystem to open/close gate.

2.3 Subsystem Requirements

2.3.1 Power Subsystem Requirements

The power line must be single phase 120 Vrms and dropped to 12 Vrms by a voltage transformer. Due to the large amount of power when transmitting Wi-Fi, we need a transformer that can handle it (185D24). After the transformer, we need a Schottky full bridge rectifier CDBHD260-G to receive a DC voltage and feed it into a filter. To receive a more consistent DC voltage, we use a multi-output buck converter to drop the voltage down to 12V and 5V. The 12V is used to power the motors and solenoid while 5V is for some sensors. In the case of a power outage, our buck converter will need a UVLO to make sure the micro does not brown out. Also, to control an LED, the IC will need a power good pin. The 12V will be passed through an LDO to reach 3.3V to power the microcontroller and its peripherals. Because the LDO will be dissipating a lot of power, we employ a heat sink.

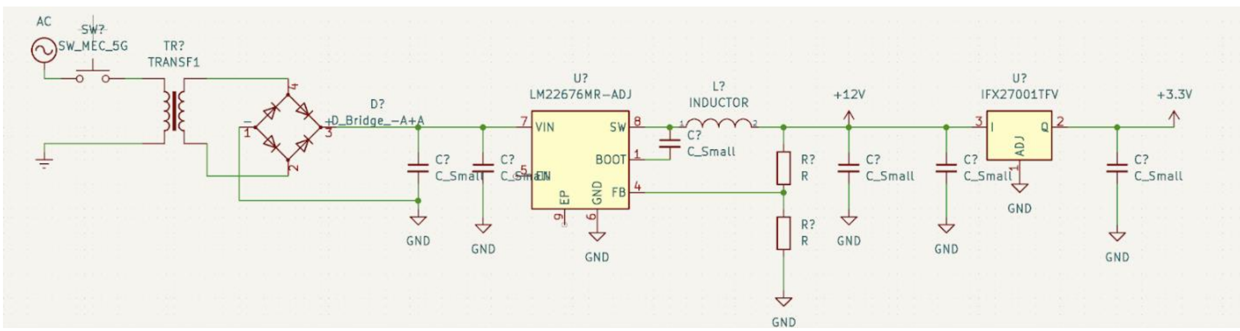


Figure 4: Circuitry of Power Subsystem

Number	Requirements	Verifications
1	Supply 12 Volts after voltage transformer.	Measure positive 12 Volts with voltmeter at the output node of the voltage transformer, with respect to ground.
2	Convert 12 Volts to 5 Volts with a buck converter.	Measure positive 5 Volts with voltmeter at the output node of the buck converter, with respect to ground.
3	Convert 5 Volts to 3.3 Volts with a low-dropout regulator.	Measure positive 3.3 Volts with voltmeter at the output node of the buck converter, with respect to ground.
4	UVLO halts circuit operation in response to power outage/decrease in power.	Measure 0 Volts with voltmeter across the rest of the circuit if UVLO is activated.

Table 1: Requirements and Verifications of Power Subsystem

2.3.2 Control Subsystem Requirements

We plan to use a ESP32 microcontroller. The ESP32 allows for a high amount of RAM and has an inbuilt WI-FI module which we will use to connect with the phone. To support the transmission of Wi-Fi, we need a large amount of current 180mA to 240mA depending on the Wi-Fi settings. The ESP32 will communicate with the sensors using SPI as the distance between the PCB and sensors is far (> 1 meter). The input voltage for the ESP32 is 2.7V - 3.6V thus our LDO must output within tolerance.

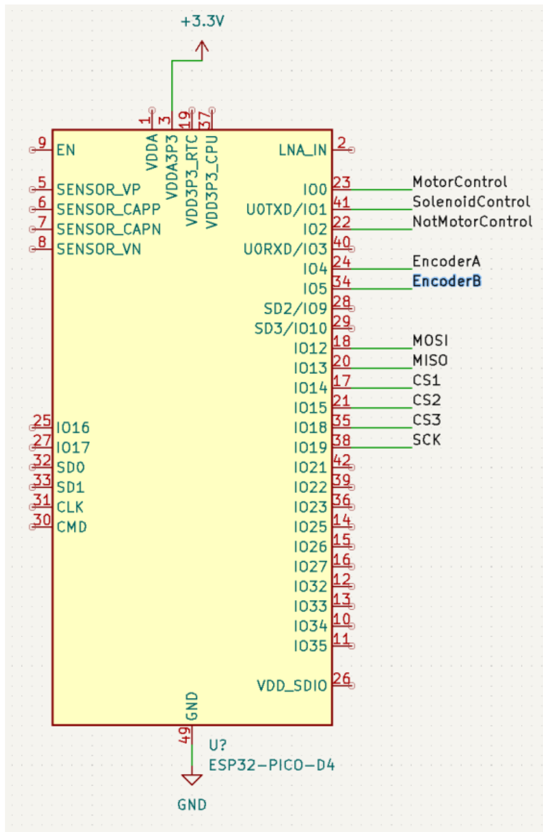


Figure 4: Microcontroller of Control Subsystem

Number	Requirements	Verification
1	Supply current between 180 mA to 240mA to Microcontroller.	Measure between 180 to 240 mA with an ammeter at ESP32 node.
2	Solenoid lock notifies the microcontroller when it is locked and the gate is in zero state (closed) position.	Check to see if the microcontroller receives data recognizing that the gate is closed.
3	LDO output within 2.7 Volts to 3.6 Volts.	Measure positive 3.3 Volts (rms) with voltmeter at the output node of the buck converter, with respect to ground.

4	Successful communication between ESP32 and sensors using SPI.	Write a program to interpret sensor readings and verify the measurements from a distance of 3 feet away from the gate.

Table 2: Requirements and Verifications for Control Subsystem

2.3.3 Detection Subsystem Requirements

Because our design requires one custom PCB, we may use a 5V PIR breakout board for motion and human detection from Adafruit. This PIR can detect up to 20 feet depending on the configuration passing our requirements. This board will be connected to the main board via the use of GPIO on the ESP32. For the proximity sensors, we will use HC-SR04 Sparkfun breakout board and convert it to the SPI interface to account for the noise from I2C. The required voltage is 5V. Finally, to be able to read the RFID, we will use RC522 that can be controlled with SPI protocol. It takes in a 3.3V input. SPI interface allows us to use less GPIO (4) while accessing many different devices.

Number	Requirements	Verification
1	Supply 5 Volts to the HC-SRO4 breakout board.	Measure positive 5 Volts with voltmeter at the HC-SR04 node, with respect to ground.
2	Supply 3.3 Volts to the RC522.	Measure positive 3.3 Volts with voltmeter at the RC522, with respect to ground.

Table 3: Requirements and Verifications for Detection Subsystem

2.3.4 Prevention Subsystem Requirements

Due to the difficulty in creating a product of size, and consulting the Machine Shop, the product is downscaled to 20 in x 20 in. The motor is used to open and close the gate and will be controlled using an H-Bridge IC (controlled by a microcontroller). An encoder is necessary for the control subsystem to measure the gate's arm position (0-90 degrees). The solenoid lock is used to ensure that the gate is safe when closed and will require 12V (Digikey 1512). Since we cannot directly control 12V, a PWM will be used for the motor to ensure smooth operation, and a Mosfet for the solenoid lock.

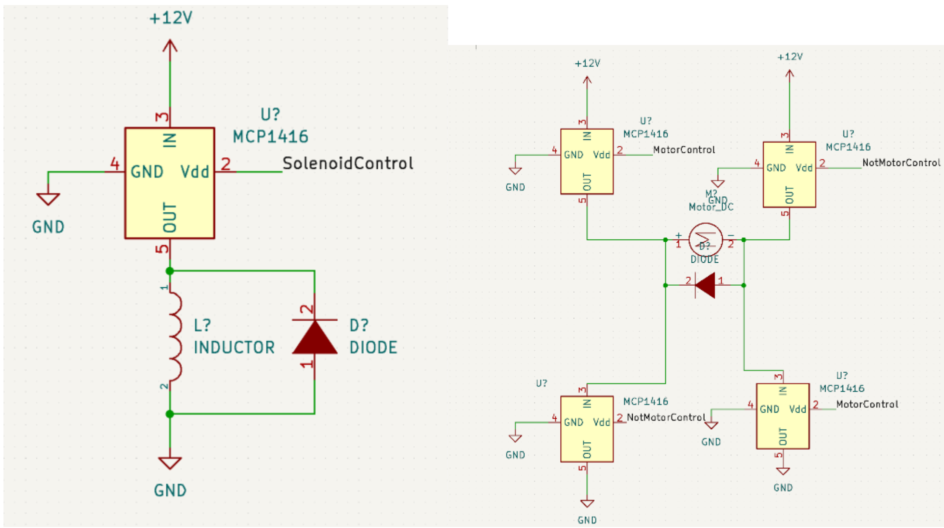


Figure 5: Circuit of Prevention Subsystem

Number	Requirements	Verification
1	Supply 12 Volts to the solenoid lock.	Measure positive 12 Volts with voltmeter at the solenoid lock node, compared to ground.
2	Gate position is capable of changing between 0-90 degrees.	Check gate position in closed and open state to ensure that the degrees are 0 and 90 respectively.
3	Gate opens and closes slowly, removing risk of hitting an object before it is detected.	Monitor and record gate movement speed to ensure it fits our requirements.

Table 4: Requirements and Verifications for Prevention Subsystem

2.3.5 Application Subsystem Requirements

The app should be easy to follow and use. It should give a notification when the door opens or closes.

Number	Requirements	Verification
1	The application should notify the user of the current status of the gate (whether it is open or closed).	Repeatedly open and close the gate and check ensuring whether the application constantly displays the correct state.
2	The user should be able to manually open/close the gate by pressing a button on the application.	Monitor whether the gate behaves appropriately when the user presses the button to open/close it.

Table 5: Requirements and Verifications for Application Subsystem

2.4 Tolerance Analysis

Due to the devices above, the tolerances of the power lines will need to be 12V±.5%, 3.3V±5% and 5V±5%. We do not care about the tolerances of the sensors are stated in the high level and will be programmed according to the aforementioned.

3 Cost and Schedule

3.1 Cost Analysis

3.1.1 Parts

Part Description	Manufacturer	Part Number/ID	Supplier	#	Unit Cost ($)	Total Cost ($)
Voltage Transformer	Hammond Manufacturing	185D24	Mouser Electronics	1	28.64	28.64
Full Bridge Rectifier	Comchip Technology	CDBHD260-G	Mouser Electronics	1	1.50	1.50
Microcontroller	Espressif Systems	ESP32	ECE Shop	1	0.00	0.00
Motion Sensor	Adafruit	189	Adafruit	1	9.95	9.95
Proximity Sensor	Sparkfun	HC-SR04	Sparkfun	?	17.95	17.95
RFID Reader	SunFounder	RC522	Amazon	1	8.99	8.99
H-Bridge Motor Controller	Texas Instruments	DRV8828	TI	1	3.573	3.573
Solenoid Lock	Adafruit	1512	Digikey	1	14.95	14.95
Buck Converter	TI	LM22676MR	Mouser	1	5.69	5.69
LDO	Infineon	IFX27001	Mouser	1	1.28	1.28

3.1.2 Labor

Labor calculations below are based on data collected from average University of Illinois Urbana Champaign ECE graduates. 98,000 USD per year.

1. Hours of Work per Week = 8 hours
2. Hourly Salary = $47
3. Total Project Duration = 11 weeks

Total Labor Cost = (Employees) * (Hourly Salary) * (Total Project Duration) * (Hours per Week)

Total Labor Cost = (3) * (47) * (11) * (8) = $12,408

3.1.3 Total Cost

Total Cost = Parts + Labor

Total Cost = $92.52 + $12,408 = $12500.52

3.2 Schedule

Week Starting	Alexander Chin	Brandon Lau	Zeyad Irsheid
(9/25)	• Design Document • PCB Design	• Design Document	• Design Document • PCB Design
(10/2)	• Prepare Design Review • Order Power Subsystem & Control Subsystem • PCB Design	• Prepare Design Review • Programming microcontroller • Verify part orders	• Prepare Design Review • Order Detection & Prevention subsystem parts
(10/9)	• First Round PCB Order • Programming Microcontroller • Build and test Power and Control System	• First Round PCB Order • Programming Microcontroller • Finalize with machine shop on mechanical design and make adjustments • Assist with building and testing subsystems	• First Round PCB Order • Programming Microcontroller • Build and test Detection subsystem
(10/16)	• Second Round PCB Order • Review PCB Design • Begin integrating subsystems	• Second Round PCB Order • Review PCB Design • Program mobile app	• Second Round PCB Order • Review PCB Design • Program mobile app
(10/23)	• Continue integrating subsystems	• Continue integrating subsystems	• Continue integrating subsystems
(10/30)	• Verify/Debug product and make necessary adjustments	• Verify/Debug mobile app and make necessary adjustments	• Verify/Debug mobile app and make necessary adjustments
(11/6)	• Full project testing	• Full project testing	• Full project testing
(11/13)	• Mock Demo Preparation	• Mock Demo Preparation	• Mock Demo Preparation
(11/20)	• Fall Break	• Fall Break	• Fall Break
(11/27)	• Final Demo	• Final Demo	• Final Demo

(12/4)	• Final Presentation	• Final Presentation	• Final Presentation

4 Ethics and Safety

We intend to abide by the regulations established by both the IEEE and ACM code of ethics.

Throughout development of the project, we intend to create a product that relates to the safety of young children. As a result, necessary precautions will be taken to ensure compliance with the IEEE ethical code 7.8.1.1 stating "[to] hold paramount the safety, health, and welfare of the public" as well as AMC code 1.2 in which we promise to "Avoid Harm" **[2]**. Regulations of these codes will be followed by running multiple tests on the sensors and motors. We will ensure that the proximity sensors and motors don't move the gate if there is a chance that anyone can be injured as a result of the movement.

In addition, since we plan on allowing the user of the gate to have an app that receives signals from the microcontroller, we will pay great attention to ensure we abide by AMC code 1.6 in which we will "Respect Privacy" **[2]**. The app will only be used to allow the user to interact with and give commands to the microcontroller. The app will not be able to access any unnecessary information on the user's device.

5 References

[1] Iyamba, N. (2012, March 12). *Stairs Among Leading Causes of Injury, Death for Kids*. KSL.com. https://www.ksl.com/article/19560857/stairs-among-leading-causes-of-injury-death-for-kids#:~:text=Every%20six%20minutes%2C%20a%20child,stair%2Drelated%20falls%20every%20year

[2] IEEE. “”IEEE Code of Ethics”.” (2016), https://www.ieee.org/ about/corporate/governance/p7-8.html