Department of Electrical and Computer Engineering
14:332:418:01 Capstone Design: Sensors, DSP, and Controls
Spring 2014

SURPASS:

Supplemented Urgency Regulating Personal Alert Safety System

Final Report

Professor: Dr. Dario Pompili
Graduate Mentor: Parul Pandey
Members: Minh Pham
Paul Kania
Matthew Brazza

Table of Contents

- Abstract
- Introduction
- Approach, Methods, and Results
  - Approach
    - 1. Design considerations
    - 2. Initial Assembly and Prototyping
    - 3. Further Prototyping and Enclosure Design
    - 4. Final Prototyping and Testing
    - 5. Wireless Integration
  - Methodology and Results
    - 1. Overall Structure and Urgency Regulating Algorithm
      - Urgency Regulating Logic
        - Table 1: Urgency Regulating Operating Modes and Danger Triggers
        - Table 2: Idle, Pre-alarm, and Alarm Times
        - Chart 1: SURPASS Function Flow Chart
    - 2. Orientation Sensing using isUpsideDown()
    - 3. Fall Detection using hasFallen()
    - 4. Motion detection using hasMoved()
    - 5. Serial Communications via XBee
- Cost and Market Analysis
  - Cost Analysis
    - Table 3: Cost of Parts for SURPASS (listed in USD)
  - Market Analysis
- Future Work
- Conclusion
- Bibliography
- Acknowledgements

Abstract

Firefighting has long proved to be a dangerous occupation. Developed in the 1980's, the Personal Alert Safety System device, better known as the PASS device, has proved to be a crucial tool for saving lives. PASS devices are distress alarms that are activated after a period of no motion, alerting others to a possible downed firefighter, even when the downed firefighter is not able to call for assistance himself. Although PASS devices have indisputably saved lives, their principle of *operation has not changed since their inception*, being prone to false activations, among other issues.

SURPASS intends to evolve the PASS alarm by focusing on several aspects of the PASS alarm. SURPASS aims to improve the basic alarm functionality by utilizing additional sensors to create a contextually aware device, one that regulates the urgency of alarm activation in order to both reduce the potential for false alarms and expedite the reaction to dangerous situations. The time required for the alarm to activate is varied in accordance to the multitude of scenarios a firefighter may encounter. By having a device that is contextually aware, potential rescuers are aware of not only the fact that a firefighter is down, but also the potential threats that they may face. SURPASS can detect whether the firefighter is standing or upside down (such as in the case that a firefighter falls through the floor with an entanglement hazard), if they're moving, or if they've fallen. Detecting whether or not a firefighter is standing up allows SURPASS to prevent accidental activations during situations in which a firefighter is less likely to be in danger, allowing the firefighter to focus on the task at hand instead of worrying about resetting the PASS device, and increases the significance of an alarm activation (rescue efforts in response to a PASS alarm are typically delayed to allow for deactivation as a result of the frequency of false activations). Detecting the other scenarios allows SURPASS to respond faster to many common dangerous situations that arise from the unstable structures and limited visibility found in structure fires, giving a firefighter a call for help when it is needed most. In addition, SURPASS integrates advanced wireless features that would allow the device to communicate with key personnel that are outside of the incident. This would allow them to be able to view a firefighter's status or past activity, or call for an evacuation.

Overall, SURPASS aims to introduce simple yet effective improvements to crucial, life saving devices. With the proliferation of affordable sensors and technology in devices such as smartphones, the modernization of the PASS device has been long overdue. SURPASS is designed to surpass the capabilities of the PASS devices that exist today.

Introduction

The Personal Alert Safety System (PASS) was first introduced to the firefighting industry in the 1980s. At the time, the PASS contained a physical motion detecting device, such as a tilt sensor. The PASS was designed to sound an audible alert after 30 seconds of no motion, or after a manual alarm button is depressed, allowing others to know of a possible downed firefighter. It also incorporates a pre-alarm signal that plays 10 seconds prior to the full alarm [1], warning the wearer to move before the alarm is sounded. Once the alarm is sounded, the user must reset the alarm using a reset sequence (depending on the design of the product, typically double clicking a button, or pressing two buttons simultaneously), as motion will no longer reset the device in alarm mode.

Legacy Equipment: PASS devices typically come in two forms; either as a standalone device or as a device integrated into a firefighter's self contained breathing apparatus (SCBA). In the case of devices integrated into an SCBA unit, the PASS is connected to a pressure switch that activates once the air supply is turned on, and prevents deactivation until all air is purged from the system. PASS devices in the US are made to the standards set forth by the National Fire Protection Association (NFPA), specifically by NFPA standard number 1982. NFPA 1982 contains performance standards that dictate everything from design to testing. New revisions typically are released every 5 years, with the most recent version being released in 2013.

Fig.1: Two typical PASS devices are shown. The device on the left is a standalone device designed to be attached to one's turnout coat or SCBA strap. The device on the right is a Self Contained Breathing Apparatus (SCBA) with an integrated PASS device. This is the type of device SURPASS is modeled after. Note the control console for the PASS being shown on the right hand strap, with the electronics in the black enclosure mounted on the hip under the cylinder. The SCBA on which SURPASS is mounted is a similar model, but without the integrated PASS device.

NFPA 1982 has undoubtedly pushed PASS alarms to be better devices, calling for more durable devices with better audible performance, and most recently, a universal PASS alarm tone. However, at the same time, the actual function of the PASS alarm has remained very much the same even after nearly 3 decades, activating after a fixed period of no motion. Hardware has since changed from the physical motion/tilt switches of the 80's to solid state triple axis accelerometers [2], but no new functions have since been added to take advantage of this new hardware, nor have there been major new ways to detect whether or not a firefighter is in a dangerous situation (although some devices have an optional temperature sensor that alerts the user to excessive heat exposure).

From a firefighter's perspective, a PASS alarm is an annoying fact of life. Although it can prove to be a lifesaver one day, PASS alarms are prone to false activation, erroneously sounding an alarm during tasks such as waiting to go inside or flowing water from a hose line or from an aerial ladder truck. These false activations present two problems. First, and most dangerously, false activations reduce the severity of PASS alarm activations as a whole. Perhaps the most fitting analogy to this would be car alarms, as nowadays, people essentially ignore car alarms due to their frequency of false activations. While PASS alarm activations, unlike car alarms, are certainly not ignored, time is typically taken to allow the person to reset their PASS alarm or to determine who has not reset their PASS alarm, time that can be crucial should a PASS alarm activation be a genuine emergency. The second issue of these false activations is that they distract the firefighter from the task at hand, especially tasks like roof ventilation with a saw or hose line operations, as both typically require 2 hands.

Our contribution: SURPASS stands for Supplemented Urgency Regulating Personal Alert Safety System. As the name implies, SURPASS aims to *improve* the old PASS alarm concept. Supplemented is in regards to the hardware found in PASS, that is, SURPASS contains additional sensors and hardware that a PASS device does not. Chief among these hardware additions is a gyroscope, which measures angular velocity. Gyroscopes are most often paired with triple axis accelerometers, a combination that is sometimes called a six degree of freedom inertial measurement unit (6DOF IMU), as they are capable of measuring both acceleration and angular velocity along 3 axes. 6DOF IMUs are most commonly found in modern smartphones, where they enable functions like screen auto rotation or motion game control. Adding a gyroscope to an accelerometer allows one to better determine orientation in space. The other hardware addition is an XBee 802.15.4 wireless radio, which sends various information to an outside command center.

The Urgency Regulating portion of the SURPASS acronym describes the algorithm and logic that takes advantage of all of the hardware. The standard PASS alarm function forces a less than ideal choice between erring on the side of caution and preventing erroneous activation. If the alarm activation period is too long, the user is given more opportunity to reset the alarm through motion during their normal activities, but the activation is delayed in an actual emergency. Erring on the side of caution by setting the activation period shorter incurs the issues of false activations that were previously explained. The Urgency Regulating algorithm is designed to determine whether the sensors suggest the user is in one of 3 conditions: Safe, Normal, or Danger. If the user is thought to be Safe, the activation of the alarm is delayed, to allow for more opportunity to reset the alarm by motion through normal activity. If the User is thought to be in Danger, the activation of the alarm is expedited, in order to call for help sooner. Normal mode acts (in terms of alarm duration) like a normal PASS alarm, containing ambiguous situations in which the safety of the user is not able to be determined. Thus, the Urgency Regulating algorithm is designed to give the user both a reduction in false activations, and faster response to life threatening scenarios.

All of this data is sent wirelessly to a computer at the command center, giving the Incident Commander information that he needs to act decisively. Furthermore, SURPASS is designed to be a platform for future life saving technologies, such as additional advanced sensors that give the Incident Commander even more information, information that might suggest that an environment may get dangerous, before that environment poses a threat to life and safety.

Fig.2: SURPASS mounted in an SCBA. There currently is no air cylinder mounted in the SCBA. Major electronic components are located in the main enclosure (red) near the hip, which is connected via a DB-9 cable to the control console. The control console is the black enclosure on the right with the silver button. It contains the manual alarm button, reset button, and status LEDs. The design choices and overall layout will be discussed in further detail later in this document.

Our Approach, Methods, and Results

Our Approach

1. Design considerations

The overall design of the unit had to be determined first before anything else could happen. As previously mentioned, PASS alarms come as either standalone devices or as devices integrated into a firefighter's SCBA. It was decided that making an SCBA integrated style device would be a better choice for several reasons. First, the device would be more stable, corresponding better to the movement of a firefighter. This is because the main electronics for an SCBA integrated PASS alarm are located on the frame of the SCBA near the hip on the back (with a control console coming off of a shoulder in the front), where it is firmly mounted to the body and corresponds with it's movement (although some SCBAs have the PASS as an entirely self contained unit elsewhere). This arrangement also allows for more physical space than a self contained or standalone unit, allowing forgiveness for the size deviations that come with prototypes, and room for additional hardware. Finally, the SCBA is intimately tied with a firefighter's wellbeing, meaning that, if time permitted, it was possible to record data such as breathing rate. The SCBA chosen was the Scott Air Pak 50, NFPA 1981 2002 edition. This unit was already in the possession of a group member, is popular among US fire departments, utilizes the aforementioned central enclosure and console setup, with generous room for electronics, and did not have a PASS device in the current configuration. It was decided that air activation of SURPASS would not be implemented, although it is a standard feature of SCBA integrated PASS alarms. This was done for feature prioritization, to prevent the permanent modification of the SCBA, and for safety reasons (SCBA units can contain upwards of 4500 PSI of air). With these design choices in mind, the hardware was then chosen, with the functionalities desired in mind (which will be discussed later). Below are the major hardware components.

Arduino Uno: The Arduino Uno is an open source hardware prototyping board that uses an Atmel AVR ATmega328 microcontroller. The Arduino is commonly programmed using the Arduino IDE using a language called Wiring, which is very similar to C/C++ (and in fact, is capable of using libraries written in C/C++). The Arduino platform is generally prized for being open source, affordable, and easy to use, especially when interfacing with hardware. The Arduino Uno was chosen in this specific situation for it's balance between size, power, and expandability, as well as the aforementioned benefits of the Arduino platform. The Arduino Uno is one of the most common Arduino boards.

Invensense MPU-6050 Accelerometer and Gyroscope: The MPU-6050 is a combination of a triple axis accelerometer and a gyroscope in one package (6DOF IMU). A unique feature of the MPU-6050 is what the manufacturer calls a DMP (Digital Motion Processor). The DMP offloads motion processing from the microcontroller to the MPU-6050, freeing resources and allowing easy access of advanced motion data. The DMP later proved to be both a helpful addition and a finicky piece of hardware. The MPU-6050 communicates with the Arduino using the I^2C bus.

Digi International XBee Series 1: The XBee Series 1 is a wireless radio based on the IEEE 802.15.4 standard. The XBee allows easy setup of wireless networks, such as mesh networks, with considerable range with relatively low cost. It is perfect for small size data transmission. It communicates to the Arduino via Serial.

Other miscellaneous hardware included RGB LEDs and piezoelectric buzzers for use as indicators, as well as switches and buttons to be used to interact with the device. A cost analysis will be available later in this document.

2. Initial Assembly and Prototyping

After the parts were received, they were assembled on a breadboard. The first goal was to recreate the functionalities of a basic PASS alarm on the breadboard. That is, the PASS alarm would use a fixed alarm activation period of 30 seconds of no movement, with a 10 second pre-alarm. The pre-alarm tone was made to conform to NFPA 1982 2013 edition, while the alarm tone was a 4 KHz solid tone for simplicity sake (NFPA 1982 calls for frequency sweeps in the alarm tone, which are harder to implement while executing other code). The code was structured such that it would be capable of having the Urgency Regulating algorithm, but it was designed such that it would behave like a normal PASS alarm. That is, the framework was laid down for the Urgency Regulating algorithm, but any function that would help it determine the urgency of the alarm (specifically, hasFallen() and isUpsideDown()) returned a constant value such that it would remain in Normal PASS alarm operating mode. Changing the return values of these functions would successfully change the Urgency of the PASS alarm operation (noted by Serial output to a computer, shortened or lengthened delay times, and different colored LED blinks), proof of concept that the Urgency Regulating algorithm would work correctly once the functions were actually implemented.

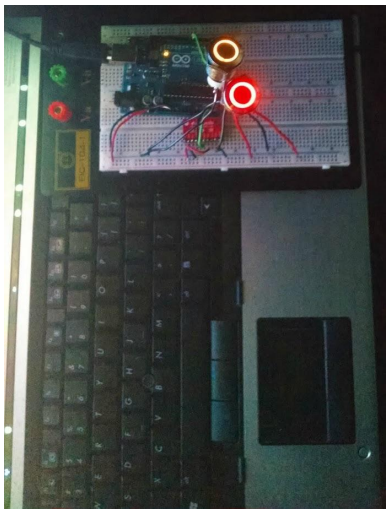


Fig.3: Breadboard prototype of SURPASS with Arduino, push buttons, and MPU-6050 visible. Piezo, LEDs, and XBee were not used at this stage. All information was instead sent to a computer over Serial via USB.

3. Further Prototyping and Enclosure Design

The implementation of the hasFallen() and isUpsideDown() functions were briefly attempted, but development was soon paused. A breadboard arrangement proved far too fragile for the testing and implementation of these functions, although isUpsideDown() was able to be implemented. A more permanent enclosure was clearly needed. Because of the unique space that the device needed to fit in, a custom enclosure was needed. The space was measured, and a design was created in CAD using Autodesk Inventor. Several iterations were created before being finalized, in order to create 3D printer friendly, ergonomic, and compact designs. The enclosures were then 3D printed by Rutgers Makerspace, using MakerBot Replicator and Replicator 2X 3D printers.

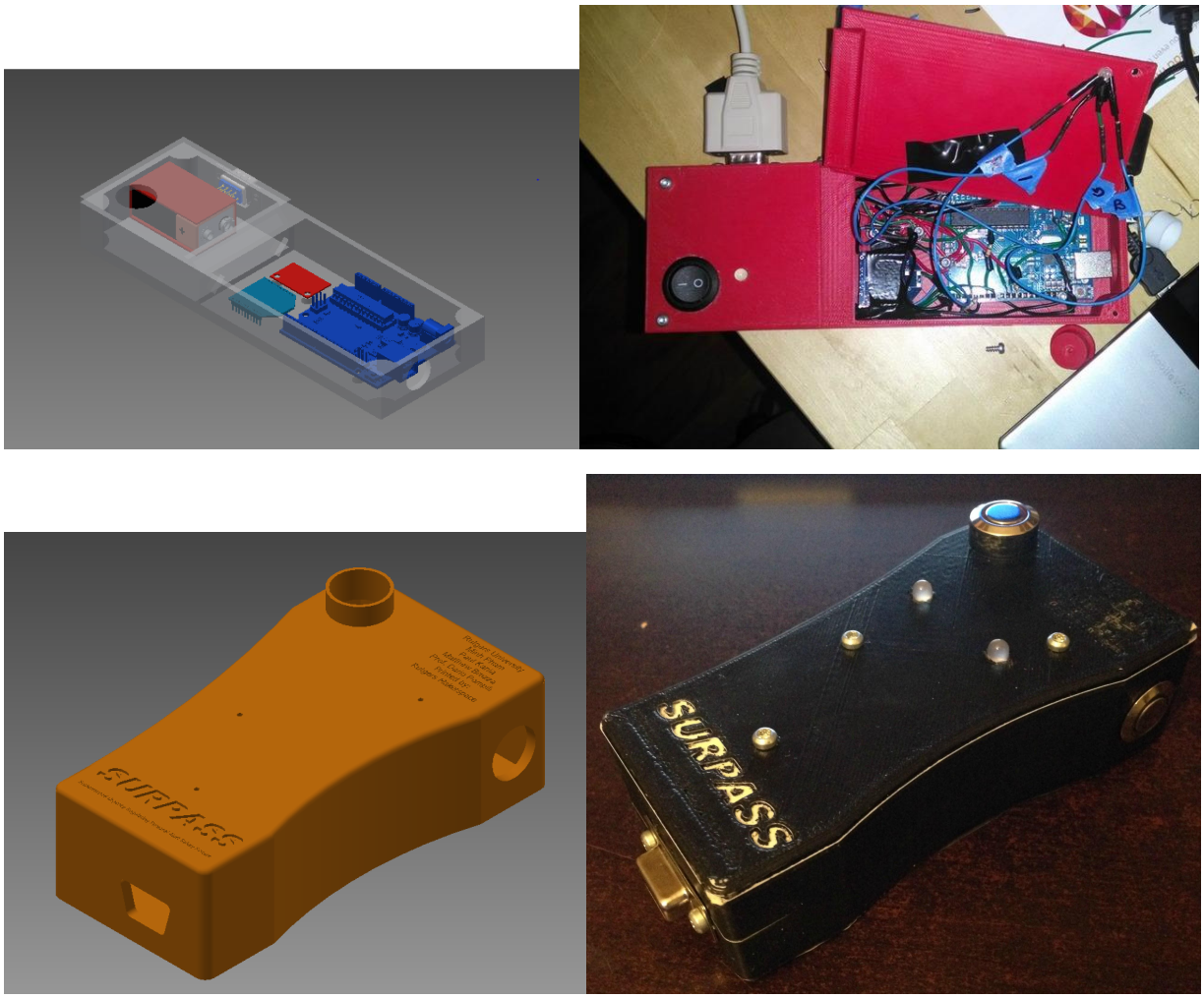


Fig. 4: Top: Side by Side Comparison of original CAD design of main enclosure and end product. Design contains two compartments. Left compartment contains DB-9 connector, battery, LED indicator, and on/off switch. Right compartment contains MPU-6050 (red), XBee (light blue), Arduino (dark blue), LED indicator, and USB pass-through (right side wall). Wire pass-throughs exist on both sides of the dividing wall.

Bottom: Side by Side Comparison of original CAD design of console and end product. Console contains Alarm and Reset buttons, DB-9 connector, and 2 indicator LEDs. Most of the engraving did not come out on the end product due to 3D printer limitations.

4. Final Prototyping and Testing

With the enclosure printed, all of the electronics were assembled, and further prototyping could occur. Free fall, via the hasFallen() function, was implemented and tested. With the SURPASS mounted, the device was worn and tested. The orientation sensing of the device was tested, ensuring that the proper operating mode was activated in various positions, including positions that are commonly assumed in the course of firefighting.

5. Wireless Integration

With the core functionality of SURPASS finished, the final task was to implement Wireless. Implementing the XBee was on the simpler side. The XBee simply communicated with the Arduino via Serial, and the device already communicated via Serial over USB to a computer for diagnostics purposes. The Serial communications were streamlined and optimized for wireless transmission (messages were made smaller, and much less frequent than before).

Methodology and Results

1. Overall Structure and Urgency Regulating Algorithm

Every Arduino program (called an Arduino Sketch) contains 2 void functions, setup() and loop(). With SURPASS, as with most other Arduino projects, run once setup code, such as hardware and variable initialization, occur in setup(), and the majority of the code lies in loop(). Specifically, with SURPASS, loop() contains:

- Timekeeping code
- Urgency Regulating logic
- Reset button polling for idle and pre-alarm
- LED blink code for idle and pre-alarm
- All pre-alarm code
- Reset button tone

All other functions are called from within loop(), specifically isUpsideDown(), which contains the orientation checking code, hasFallen(), which contains fall detection code, hasMoved(), which contains the code to check if the user has moved, fullAlarm(), which contains all the alarm code, and resetAll(), which is the reset function. There are 2 interrupts, one which is used for the manual alarm button, and one which is used for the MPU-6050, which triggers an interrupt when data is ready. IMU features utilize a library [3] that accelerated development of these functions.

Urgency Regulating Logic

A major component of SURPASS is the Urgency Regulating algorithm. Simply put, the Urgency Regulating algorithm makes SURPASS alarm sooner in dangerous situations, and allows SURPASS to alarm later in safe situations.

The Urgency Regulating code calls hasFallen() and isUpsideDown() to determine how much danger the user is in, which in turn determines the time it takes for SURPASS to sound the pre-alarm and alarm. The details of what triggers each danger value, and how each danger value affects the behavior of SURPASS is shown below.

Table 1: Urgency Regulating Operating Modes and Danger Triggers

Operating Mode	**Safe**	**Normal**	**Danger**
Danger Value	0	1	2
Danger Trigger	Standing Position	Horizontal Position	Upside Down Position, Fall Detected
Firefighting Scenarios	Standby prior to work, Aerial Waterway Operations, Hoseline Work (crouching or standing), Forcible Entry, Overhaul, Ladder Work	Crawling/search and rescue, roof ventilation*, incapacitated	Fall, floor or structure collapse, entanglement hazard on the lower body, inverted on rope line

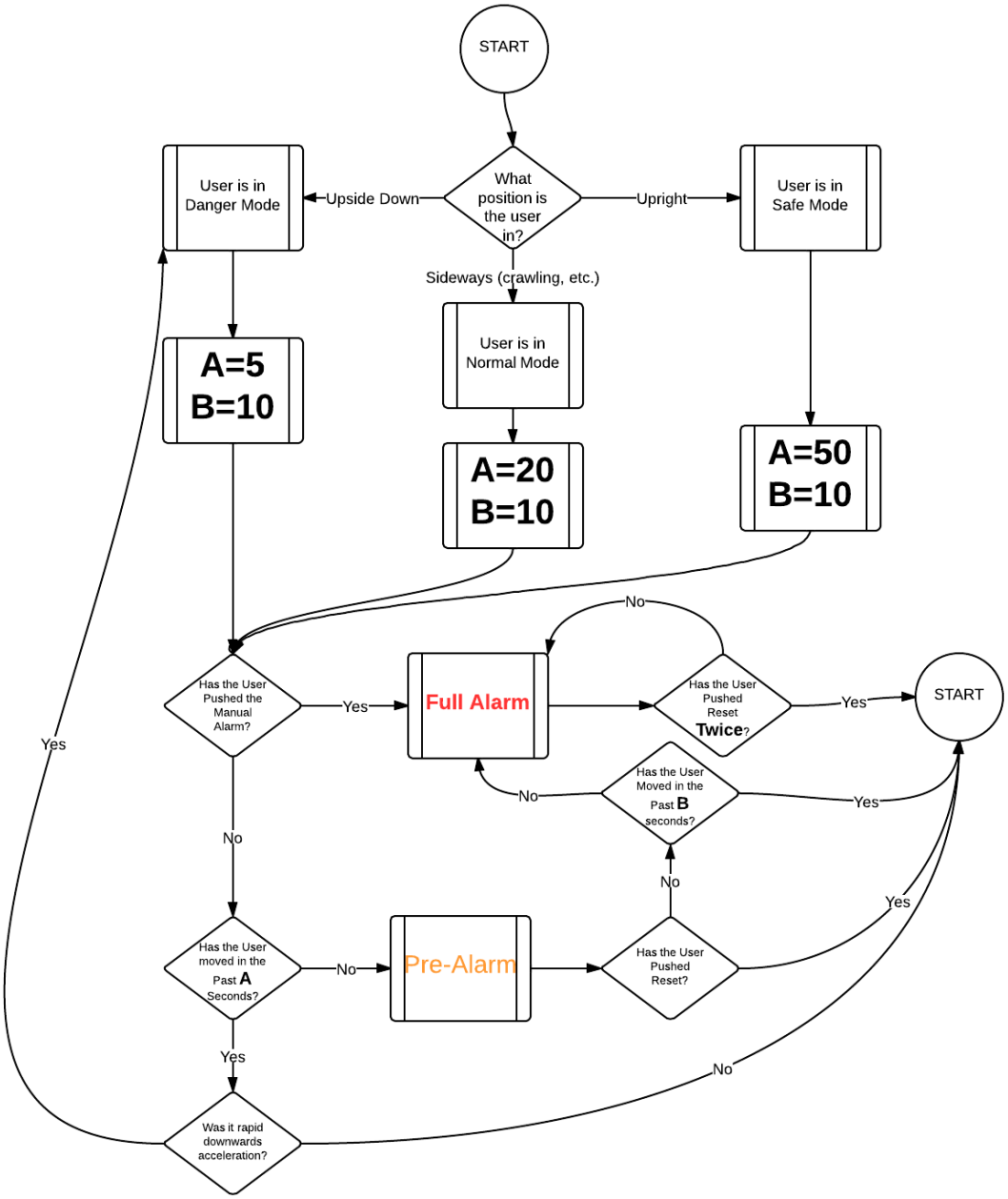
*Can be adjusted (see *2. Orientation Sensing using isUpsideDown()* for more information)

Table 2: Idle, Pre-alarm, and Alarm Times

Operating Mode	**Safe**	**Normal**	**Danger**
Danger Value	0	1	2
Idle Time before Pre-alarm	50 s	20 s	5 s
Pre-alarm time	10 s	10 s	10 s
Alarm Time	Until Reset	Until Reset	Until Reset

Note that there are both legitimate firefighting tasks and dangerous situations in which one would be in Normal Mode (in a horizontal position), as highlighted by Table 1. This is the primary reason to have Normal Mode, to account for ambiguous contexts that have no clear danger or safety. Contrast this with Safe and Danger mode, in which almost all firefighting scenarios are either safe or dangerous, respectively. Of course, it is impossible to account for every firefighting scenario. It is very possible to be incapacitated in an upright position (triggering Safe Mode), as well as it is possible to be crawling down a steep incline that (triggering Danger Mode). This is why each mode lengthens or shortens the delay, instead of deactivating the alarm outright or immediately triggering the alarm.

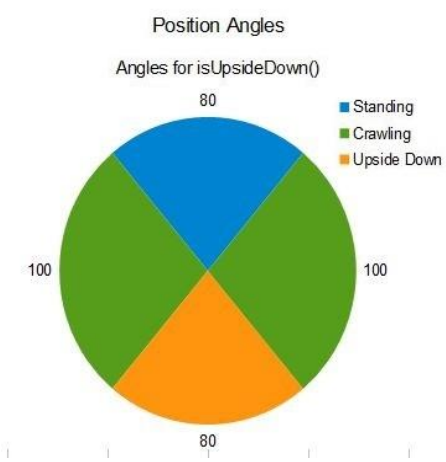
Chart 1: SURPASS Function Flow Chart



2. Orientation Sensing using isUpsideDown()

Contrary to the name, isUpsideDown() returns more than whether or not the user is upside down. It determines orientation overall. As previously mentioned, the MPU-6050 has a DMP, which offloads motion calculations from the microcontroller to the MPU-6050. The DMP is capable of outputting a large variety of data, the most crucial for isUpsideDown() being Yaw, Pitch, and Roll output. Because of the way the MPU-6050 is mounted, the value of greatest concern is Roll, which corresponds to bending forward or back at the waist. It should be noted that the Yaw, Pitch, and Roll output is susceptible to gimbal lock, a phenomenon in which a degree of freedom is lost. However, no issues were exhibited during testing, and if issues arise, they can be defeated by interpreting quaternion output instead of yaw, pitch, roll output.

isUpsideDown() uses a library [3] that obtains the Yaw, Pitch, and Roll reading from the DMP, then compares the Roll to a number of constants. If the user is within 40 degrees in either direction of standing straight up, isUpsideDown returns a value of 1, which indicates that the user is standing or crouching. If the user is within 40 degrees in either direction of being straight upside down, isUpsideDown() returns 2, which indicates that the user is upside down. Otherwise, isUpsideDown() returns 0, which indicates that the user is horizontal.



Left: Diagram of angles and their corresponding positions for isUpsideDown().

Right: Code for determining what state to apply given the inputted angle.

```
if (ypr[2]* 180/M_PI>50) {
    currentposstate = 2;
} else if (ypr[2]*180/M_PI<-50)
{
    currentposstate = 1;
} else {
    currentposstate = 0;
}
```

isUpsideDown() and the threshold values in general proved to be very effective. The tolerances included allowed for some deviation from standing straight or hanging upside down straight, such as arises out of differences in posture and differences in how the SCBA is worn. One notable exception is how SURPASS reacts when the user is in a position to user a saw to cut a roof. Such a scenario involves standing, but the user is bending over to cut the roof on which he stands. In testing, this bending over was sufficient to activate Normal Mode instead of Safe Mode. However, this testing was limited. This scenario was only tested on a flat surface, instead of a sloped roof that is more common in roof ventilation operations. Sloped roofs may limit bending by bringing the cutting surface closer to the firefighter, and by counteracting the bend through the inherent angle of the roof. Thus, it is plausible that roof ventilation operations on a standard sloped roof may still activate Safe Mode. In the case that it does not, threshold values can be tweaked until Safe Mode is activated in this scenario, taking care to avoid tweaking it too much as to cause undesired activation of Safe Mode.

3. Fall Detection using hasFallen()

The second function that allows the Urgency Regulating algorithm to work is hasFallen(). hasFallen() determines whether or not the user has fallen, using a fall detection algorithm applied to gravity compensated accelerometer data. The MPU-6050 is actually capable of generating a free fall interrupt using the DMP, but the interrupt proved to be unreliable and overly sensitive.

Once again, data was obtained using the DMP, specifically the gravity compensated acceleration values (acceleration values that have the effect of gravity removed). The values are compared to threshold values. If the values exceed the threshold, they qualify as possible free fall data samples, which cause a counter to increment. Each qualifying sample increments the counter, while a non qualifying sample decrements the counter. If the counter exceeds a threshold, hasFallen() returns accordingly.

hasFallen() is not capable of working all on it's own however. There are some tweaks that had to be made to the main loop() for hasFallen() to correctly trigger Danger Mode. It was observed that a falling object does not simply fall and hit the ground. Rather, a falling object bounces after it lands, which proves to be a problem for hasFallen(), since the movement from bouncing would reset the Danger Mode that hasFallen() is intended to trigger. To create a literal "debouncing" function, hasFallen() is designed to disable hasMoved() for 2 seconds immediately after working to prevent hasMoved() from resetting Danger Mode during any bouncing that occurs after falling.

4. Motion detection using hasMoved()

A crucial feature of any PASS alarm is the ability to detect movement, in order to be able to determine when to start the countdown to the pre-alarm and alarm. The same is true for SURPASS. hasMoved() is very simple. It obtains accelerometer readings from the MPU6050, then compares them to baseline values. If the readings exceed the baseline values by a certain threshold, hasMoved() returns true. Once it returns true, the main loop() will call the resetAll() function, which resets all of the variables, including the timekeeping variables.

5. Serial Communications via XBee

Another feature of the SURPASS alarm is the ability to communicate wirelessly with a central base. The system uses a pre-programmed XBee module to send data to/from the arduino with IEEE 802.15.4 specifications to another Xbee module. The system sends the data serially through the transmission and receiver pins of the arduino. The second Xbee is currently attached to a computer via a serial USB port and displays the data via a GUI; the system can be expanded to have information sent to the XBees of other SURPASS alarms.

Cost and Market Analysis

The one of the goals with SURPASS was to create a realistic prototype with some form of commercial viability. There are some challenges, which will be discussed below.

Cost Analysis

Below are the market prices paid for the major components of SURPASS

Table 3: Cost of Parts for SURPASS (listed in USD)

Component	**Price (Each)**	**Quantity**	**Total**
Arduino Uno R3	29.95	1	29.95
MPU-6050 Accelerometer and Gyroscope	39.95	1	39.95
XBee Series 1	28.95	2	57.90
XBee Explorer Dongle	29.95	1	29.95
Metal Momentary Pushbutton	4.95	2	9.90
Common Cathode Diffused RGB LED	1.95	4	7.80
Piezo Buzzer	1.50	1	1.50
Female DB9 Connector	1.87	2	3.74
9V Snap Connector	1.25	1	1.25
SPST Rocker Switch	0.50	1	0.50
		Total:	**182.44**

It should be noted that there are several costs not included in the above table. Shipping costs are not included, nor are the costs related to the enclosures or related screws and hardware. The enclosures were printed for free by Rutgers Makerspace, though the cost of plastic used is

approximately $6. The wires, cables, and resistors were borrowed from Rutgers. Finally, the cost of the SCBA is not included, though the SCBA itself is not important to the device.
The above table of parts should not be considered a strict guide to the cost of producing SURPASS as a commercial product. Of course, there will be a large discrepancy in costs between the prototype model and a commercial model that will arise out of the economies of scale of purchasing large numbers of components for production. A major reason to believe that the true cost will be different is that the components must be different.

An Arduino itself would not be used in the production version. Instead, a more reliable, purpose built microcontroller would need to be used. All of the electronics would most likely be assembled on a printed circuit board. A louder, more durable piezo would have to be used, and most likely more than one. The system would need to be designed to be intrinsically safe, in order to reduce the risk of igniting an explosive atmosphere. Durability would have to increase. Instead of a 3D printed chassis, a more rugged polymer chassis of some sort needs to be used. Electronics are typically potted in PASS devices for further durability.

A large amount of costs will also come from testing and certification. As previously mentioned, the device must be certified intrinsically safe, and thus must undergo testing by Underwriter's Laboratories. Durability must be tested, testing that is often inherently destructive (that is, multiple devices will be destroyed during testing). Finally, more thorough testing will need to be performed as to the effectiveness of SURPASS on the fireground.

With all these variables involved, it is hard to estimate the true cost of producing SURPASS commercially from the cost of the parts purchased. Perhaps a better way to estimate the true cost is to look at what hardware SURPASS has added to a standard PASS alarm and estimate the cost accordingly.

In reality, SURPASS only added one major component to a standard PASS alarm, specifically the gyroscope. It is unfair to say that the XBee radio is an added cost, since it is possible to create a wireless-free version of SURPASS, and because some PASS alarms, though uncommon, do have wireless capabilities. Thus, the best way to analyze the cost of SURPASS compared to a standard PASS device is to analyze the cost of adding a gyroscope to a standard PASS device.

As the table above indicates, the MPU-6050 costs $39.95 from Sparkfun Electronics, whereas standalone accelerometers range from $9.95 to $17.95 on Sparkfun. However, these are prices for breakout boards. As previously mentioned, everything would be produced on a PCB, so it would be more accurate to look at prices for the ICs themselves. In that case, the MPU-6050 IC retails for $12.95 on Sparkfun, whereas the MMA8452Q IC retails for $2.95, leading to a difference of $10. All of the previous caveats about specialized hardware, testing and economies of scale should still be considered, but even then, the price difference should not be astronomical.

$10 on it's own already sounds like a small cost, but it should be considered with the cost of a PASS device in mind. While data on the cost of manufacturing a PASS alarm is not available, one can look at retail prices to determine the impact of this addition on the cost. The Fire Store lists the cost of a standalone SuperPASS device without temperature sensing as $474.99. One should also examine the cost of an SCBA integrated PASS device, since this is the original intended application of SURPASS. According to the 2012 New Jersey State price list for Scott Safety products [4], the list price for an SCBA console without a PASS device is $1295. The cost for a console with a standard PASS device is listed as $2110, with a PASS device and the Pak-Tracker firefighter locator $2265, and with a PASS device with the SEMS II (RFID accountability and wireless) system is $2750, meaning the retail cost of adding a PASS alarm to an SCBA ranges from $815 to $1455. This means that, if a manufacturer wanted to break even with the addition of a gyroscope to a PASS device, PASS alarm prices would have to increase (using the previous $10 figure) anywhere from 0.69% to 2.11%.

In conclusion, adding the capabilities of SURPASS to standard PASS alarms would add roughly $10 in hardware, which equates to 0.69% to 2.11% extra on the retail price of various PASS devices, assuming the manufacturer would break even on the addition. It is clear that SURPASS is *extremely economically viable*.

Market Analysis

The above cost analysis suggests that SURPASS is economically viable. However, this should not be the only consideration when it comes to producing SURPASS commercially, as SURPASS faces other market barriers.

The biggest barrier to producing SURPASS exists in regulatory and standards issues. As previously mentioned, NFPA 1982 is the standard on PASS devices, and currently states that PASS alarms must alarm after 30 seconds of no movement. This means that SURPASS, with the current NFPA standard, would NOT be NFPA compliant. SURPASS may also face additional government regulation from OSHA and NIOSH.

Because there is no market for non-NFPA compliant PPE in the United States, NFPA 1982 must change to accommodate SURPASS, either as an option or as the standard, in order for it to be commercially viable. The most recent version of NFPA 1982 is from 2013, and the standard is typically revised every 5 years, meaning that SURPASS would not see the market until 2018, at a minimum. However, that gives time for further testing and development, and for proposal of the concept to the Electronic Safety Equipment Committee that is responsible for NFPA 1982.

Being designed to be mounted to an SCBA also helps the marketability of SURPASS in some aspects. SCBAs are high cost, durable goods (like automobiles), typically purchased in large quantities only once every 10-20 years. Although this infrequency of purchase makes the

marketability sound bad for SURPASS, for those in the market to purchase an SCBA, the high cost of SCBAs makes it easy to justify a small percentage increase in price for the enhanced capabilities of SURPASS, especially since the devices will not be replaced for a long time (a manifestation of the idiom “buy once, cry once”).

Future Work

SURPASS is designed to be a platform for future technology. There is a plethora of significant, life saving additions that could have been added to the device, but were not due to time constraints. Many of these ideas are also economically viable, having little to no new hardware.

Perhaps the most important, and most challenging, addition would be localization. This sentiment was even expressed by the Chief of Community Volunteer Fire Company, who expressed a desire for a GPS-like system for finding firefighters. The current solutions on the market for finding downed firefighters are more akin to playing “Marco Polo” in a structure fire, something that is less than ideal in a situation where time is of the essence. PASS alarms themselves are used to find firefighters, as the Rapid Intervention Crew (RIC) tasked with finding the downed firefighter would attempt to determine the direction of the sound emitted by the PASS device. The other solutions consist of an emitter embedded in the PASS device, and a handheld device that can be used to determine in which direction the emitted signal is the strongest. These technologies are typically RF or ultrasound based [5], and certainly are an improvement over listening for a PASS alarm, but they still aren’t able to instantly pinpoint where the firefighter in question is.

Numerous challenges exist in the implementation of a localization system. GPS does not typically work indoors, and the indoor layout of houses is typically not known beforehand, though this is changing with the prevalence of Mobile Data Terminals, on which blueprints of houses may be available. The strongest solution, it seems, is localization via wireless Received Signal Strength Indicator (RSSI). This requires no additional hardware, since SURPASS is already equipped with an 802.15.4 radio. There is research [6] that demonstrates that such localization is possible, though this method required “mapping” the room by determining RSSI at various points in the structure, something that is not possible in an emergency situation. It may be better to modify this localization system by removing the mapping requirement altogether, relying on RSSI alone for a more rough location of the firefighter in need (nodes would have to be placed in areas like the corners of the structure for reference points, which can be done when the fire department arrives). RSSI can also be used to determine the nearest firefighter, so that the nearest firefighter can assist, or report his location in an attempt to locate the downed firefighter.

An easier addition would be enhanced sensor capabilities. Breathing rate could easily be added, since SURPASS is designed to be integrated in the breathing apparatus, and could provide information on the health or state of a firefighter. Temperature sensing is found on some PASS

alarms. Combined with the previous localization addition, it may be possible to map hot spots onto blueprints of a structure, helping incident command determine problem areas. Gas sensors may be added as well, to identify additional hazards (poisonous gases, low oxygen, or explosive atmospheres), though the periodic calibration required makes such an addition less appealing.

Final additions may include RFID accountability and enhanced Heads Up Displays. Some PASS systems have RFID accountability systems, but require the user to manually scan his ID. The ID should instead be strategically embedded in one's turnout coat. The reader should likewise be strategically embedded in the SCBA, for example, in the back plate area. Once the user dons his SCBA, SURPASS will be able to automatically read his ID tag, and identify the user on the incident commander's mobile data terminal. SCBAs have also have heads up displays (HUDs), which consists of LEDs that indicate to the user how much air he has. The capabilities of SURPASS should be integrated into these HUDS, showing the user his current status. Combined with the previously mentioned ideas like localization and enhanced sensors, the user may be able to easily locate the nearest firefighter or determine the current environmental conditions.

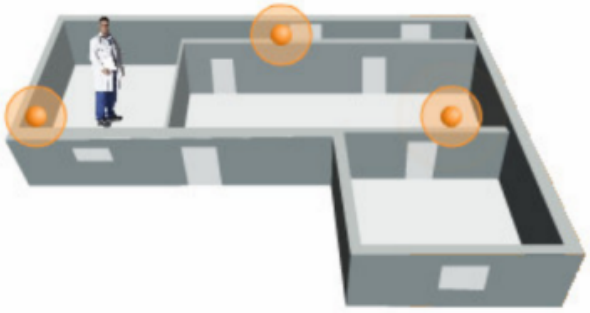


Fig.5: Left: Handheld RF Receiver used in some firefighter locating units. Receiver indicates direction of strongest user reception. Right: Indoor Localization System proposed by Pascual. A modified version may be possible with SURPASS.

Fig.6: Left: Standalone gas detector. Integrating gas detection into SURPASS can improve safety and situational awareness, though doing so may increase costs and require periodic calibration. Other possible sensors include temperature and breathing rate.

Middle: RFID card. If properly embedded in protective clothing, accountability/identification can be automatic with no action required on part of the user.

Right: Heads Up Display for SCBA. Enhancing the Heads Up Display with information from SURPASS (of both current and future features) can improve situational awareness.

Conclusion

PASS alarms have existed for more than three decades. In that span of time, technology continued to get more and more complex, but the device was never changed. There are millions of people on this planet with handheld devices that can do more than the PASS alarm. As firefighters risk their lives and put themselves in hazardous situations to save others, it goes without saying that the equipment they have should be using the best possible technologies and not 30-year-old hardware.

SURPASS was designed with the firefighter in mind, making it more contextually aware than the current PASS alarm, though with changes that should've been made years ago. SURPASS uses an accelerometer like the PASS alarm does, but also uses a gyroscope to allow it to be more aware of the orientations of the firefighter. This feature alone would be marketable enough to change the PASS alarm. To ensure the appropriate communication of the data we collect, SURPASS also uses wireless radios to let people outside of the incident know the exact situation that a certain firefighter is in; e.g., the incident commander can direct people with a greater knowledge of the operation. Additionally, since SURPASS uses an Arduino, the addition of sensors would be very easy to do, allowing SURPASS to further expand in a very simple way.

Overall, looking at the PASS alarm in comparison to the SURPASS device that was made, it's really shocking that this technology hasn't been incorporated into the PASS device yet. Taking the idea of the PASS alarm and pairing it with the technology of today gives a very valuable product that can seriously impact the greater safety of firefighters and those involved.

Bibliography

[1] NFPA 1982, Standard for Personal Alert Safety Systems (PASS) (National Fire Protection Association 2013)

[2] Grace Industries, "SuperPass 3" from
http://graceindustries.com/index.php/public-safety-man-down-alarms/29-superpass-3

[3] Jeff Rowberg, "i2cdevlib" from https://github.com/jrowberg

[4] Scott Safety Pricelist, 2012 from
http://www.state.nj.us/treasury/purchase/pdf/t0790/Line42-ScottSafety-ScottHealth.pdf

[5] ISI EchoTracer, from
http://www.summitsafetyinc.com/products/isi/EchoTracer_datasheet_4-3-09.pdf

[6] Pascual, Maria D.G., 2012, "Indoor Location Systems Based on ZigBee Networks"

Acknowledgements

We would like to acknowledge the **Community Volunteer Fire Company and Franklin Township Fire District #3** for letting us use their facilities and equipment. Also, we would like to thank Dr. Dario Pompili and our graduate mentor Parul Pandey for their guidance. Lastly, we would like to thank **Rutgers Makerspace** and Rick Anderson for 3D printing our enclosures.