Final Project Report (Dv3)

SmellOvision

"Augmenting the audiovisual experience"

Due: April 6th 2019 - 17:00

Group #16

Matthew H. MacLennan

Joshua Bell

Preface 3

Document Overview 3

Terminology 3

Problem Statement 4

Summary of Proposed Specifications of the Design 4

Principle of Operation 5

Performance 5

System Characteristics and Constraints 5

Acceptance Test: 6

System Architecture 6

User Input 7

Mechanical user Interface: 7

Personal Computer: 7

FRDM-K64 8

Amplifier 8

Spraying Mechanism 8

Aerosol Can 8

Output 9

System State Diagram 9

System Design 10

Hardware Design 10

L.M 50 kΩ Potentiometer Voltage Divider 11

Power Amplification 12

Geared DC Motor 13

Spraying Mechanism 14

Software Design 15

Hardware Interface 16

FRDM-K64 Interface 16

Java Interface 20

Term Test Plans 20

Evaluation 24

Acceptance Tests 24

Current Status 25

Future Work 25

User Manual 26

References **30**

Preface

Mr. Chuck Roherty and Mr. Vivek Sabbarwal,

Within this final document you will find the relevant specifications of the prototype your group requested for. This prototype is the enhanced multimedia device with an added sense for the user to experience, where that extra sense is smell. If you need any clarification regarding the content of this document, please do not hesitate to reach out to us via email.

Thank you both for allowing us to have the opportunity to work with your group.

Respectfully yours,
Matthew M. - mmaclen2@unb.ca
Joshua B. - joshua.bell@unb.ca

Document Overview

This document encompasses all components relative to the final prototype that was designed to meet all of your requirements and specifications set in the contract. The document includes terminology, the specifications set for the product, system architecture and design, hardware design and specifications, software design and specifications, prototype specifications and operation, testing, current state evaluation and a user manual.

Terminology

This section is merely to provide a quick and straightforward breakdown of various terms that will be used throughout the document. This section is not hardware or software specific; however it will outline terminology used that could potentially be misleading.

The Viewer:

"The Viewer" is the primary individual who interacts with the system. Please note that "The Viewer" and "The User" can be used interchangeably.

The System:

"The System" refers to the amalgamation of all individual components of the functional prototype. This includes the user interface, aerosol can, additional hardware and the casing.

The User Interface:

"The User Interface" (U.I) refers to the two points of interaction between the user and the system. They are the "Mechanical User Interface" (M.U.I) and "Graphical User Interface" (G.U.I).

Mechanical User Interface (M.U.I):

The M.U.I refers to the mechanical components are require the user to interact with. These components are the adjustable knob, push button and USB C port. The purpose of these components will be outlined in future sections.

Graphical User Interface (G.U.I):

The G.U.I refers to the software application developed in Java that must installed the personal computer that will be displaying the video to the user.

The Board:

"The Board" refers to the FRDM K64 board given to the design group at the beginning of the semester. This will only be used in 'high-level context'. Further breakdowns of individual ports, modules, pins, etc. will be done in future sections.

The Spraying Mechanism:

"The Spraying Mechanism" is the developed mechanical hardware that is responsible for administering the fragrance. This is composed of a stand, motor mount and shaft attachment that creates the appropriate downward force to administer roughly a 250ms spray. Documentation and illustrations of this device will be elaborated on in future sections.

Problem Statement

Design, build and test a working prototype that will compliment a viewer's experience while watching a video. This must be accomplished by releasing a fragrance at the appropriate time, which designated by the viewer. Using a simple adjustable knob, the user should be able to decide how much fragrance is dispersed during the video.

Summary of Proposed Specifications of the Design

The following section is a summary of the past communications that took place earlier this term. Below you will find the proposed principle of operation, performance metrics and finally system characteristics and constraints.

Principle of Operation

The board will be turned on when the user plugs the USB C port into the board and the other end into their personal computer (PC). The user will open the application installed on the PC. The G.U.I will appear - allowing the user to select which MP4 file they wish to play and the time the fragrance to be dispersed. (Note, the fragrance will be dispersed over a five second 'window'.) Afterwards the user will rotate the adjustable knob to select the number of sprays administered within the window. The user will then press start on the G.U.I and view the video at their leisure. If they wish to pause the video and spray countdown, the user can accomplish this be pressing a switch. Once both the fragrance is administered and video is finished, the user may decide if they'd like to turn the device off by unplugging it or can select the next video to be played (as well as reset the timer to a new desired value).

Performance

The developed application must be able to accept MP4 files and display them properly. The fragrance must only disperse at the time selected by the user. This delay must be at least 1 second and can increment by an integer number of seconds. The fragrance will only be dispersed over the 5 second window at the rate inputted initially by the viewer. The rate of sprays must fall between 0-5 times per window.

System Characteristics and Constraints

- The only connection allowed is a single USB C cable to connect the viewer's PC and the System.
- There must be a pause/play button located on the board and is accessible to the viewer.
    - This must cause the internal timer to stop.
- The delay must be selected by the user via the G.U.I.
    - Minimum of 1 second.
    - Must be an integer value.
- The 'window' at which the aerosol can can be pressed must be 5 seconds long.
- The aerosol can must be pressed at least once over the available window and up to a total of 5 times.
- Any additional power supply must be provided by replaceable batteries.
- The system must ensure that throughout the viewing the application does not disturb the experience. Such as:
    - No buffering or delays over 1 second.
    - No sound or extremely noticeable drops in video quality.
- The total cost of the system must be less than $45.00.

- ❖ The system must be enclosed within a casing that has the following properties:
  - ➢ Large enough for a Macbook Laptop to sit comfortably on top.
  - ➢ Stable.
  - ➢ The casing and internal components should have a combined weight of less than 15lbs.
  - ➢ Additional features but are not mandatory would include:
    - ■ Carrying straps.
    - ■ Compartment located internally to store difference fragrances.
  - ➢ All features of the M.U.I must be labelled and easily accessible.
  - ➢ The opening at which the fragrance comes out of the casing must be appropriately labelled.

Acceptance Test:

A prototype will be accepted based upon the demonstration of the system properly performing the stated requirements. This will include:

1. Acquiring user input via the functional M.U.I and G.U.I.
2. Proper video playback without any hindrances as listed in the System Characteristics and Constraints section.
3. Functional pause/play button.
4. Appropriate window over which the selected scent is administered.
5. Appropriate amount of fragrance is released.
6. The entire procedure is straightforward for the viewer.

System Architecture

Below is a system architecture block diagram and a state flow diagram of the system created to meet the specifications of the design. It should be noted that this section is simply an overview of the approach our firm took and thus in the following sections the implemented topology will be broken down and discussed on a lower level.

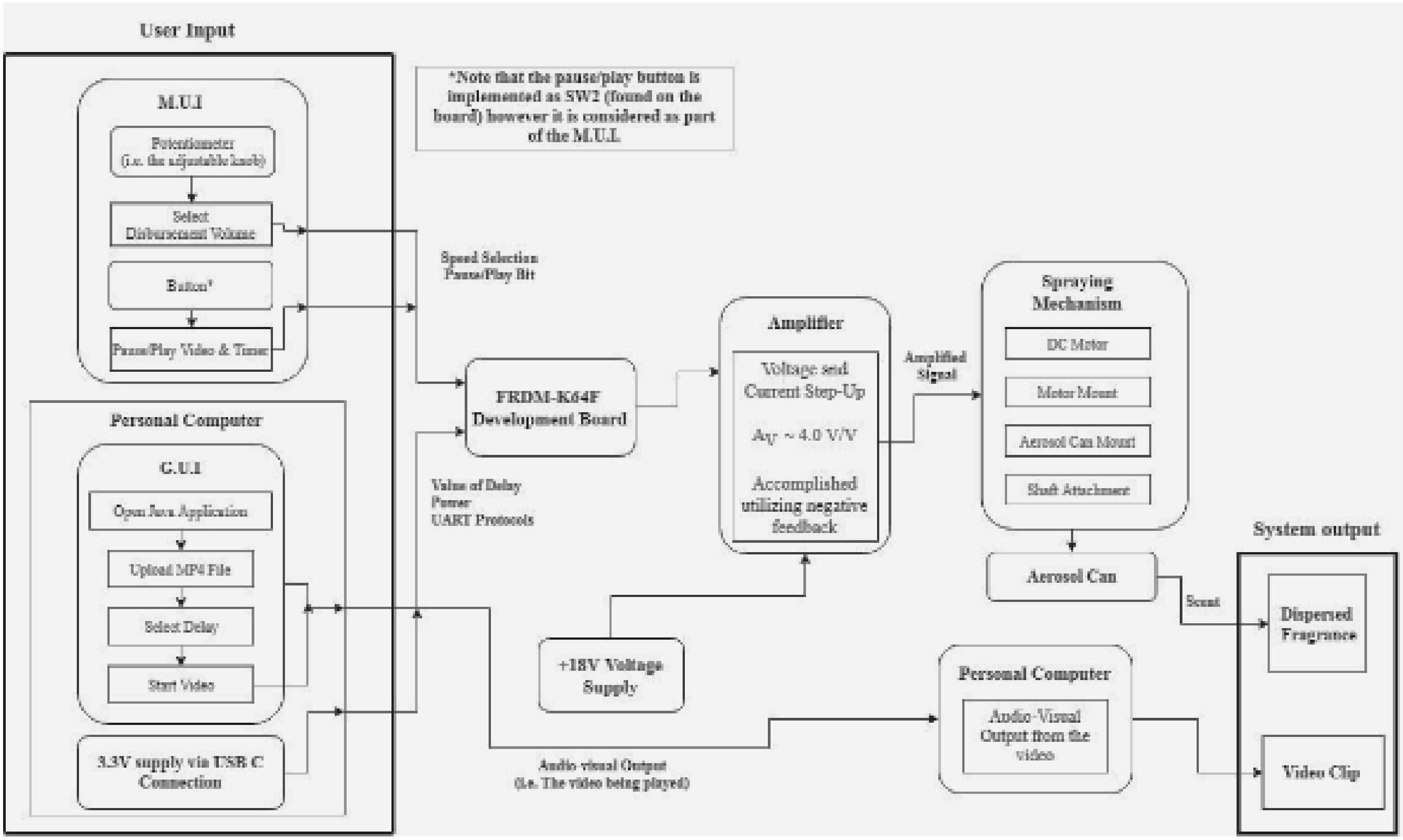

Figure 1: System Architecture

User Input

Mechanical user Interface:

The mechanical user interface is where the viewer make select the amount of fragrance they wish to administer during the video being played. This is accomplished by a potentiometer that will adjust the input voltage to the board. The available is able to pause/play the video after the start video option is selected on the G.U.I in case the user must temporarily divert their attention.

Personal Computer:

The personal computer is firstly used as the power supply to the FRDM board. This is supplied using a USB C cable that will provide ~3.3V and the necessary current for operations. The G.U.I is where the viewer may select and upload their video to the application (must be an MP4 file). As well, the application will then allow the user to select an integer number of seconds, this will be the delay that will take place between the start of the video and the time where the scent is administered. After both the video and delay are inputted, the G.U.I will prompt the user to start the video.

FRDM-K64

The FRDM-K64 is responsible for processing the various inputs and managing the appropriate output throughout the entire operation of the system. The board will receive the input voltage from the potentiometer and will store the value until needed. The board processes pause/play requests made by the user once the video is started and utilizes the internal clock to appropriately count to the requested delay before administering the output signal to the amplifier.

The board is also communicating with the Java application via UART. Utilizing this connection, the application is able to send the inputted data (delay) as well as receive commands from the board (pause and play).

Amplifier

The amplifier is a simplistic approach to ensuring that the spraying mechanism (primarily the DC motor) has both enough current and voltage. This was accomplished by utilizing a power op amp with a negative feedback network and +18V supply voltage to amplify the outgoing signal from the FRDM board. This enabled the motor to have enough power to operate.

Spraying Mechanism

The spraying mechanism is the device that is responsible for applying a downward force onto the selected aerosol can. This involves a stand that holds the aerosol can as well as the motor mount. The mount is positioned close to the push button of the can, and with the motor's shaft attachment, the motor is able to rotate and temporarily apply enough pressure to dispense the fragrance from the can for less than a quarter a second (in other words - a small spritz).

Aerosol Can

This is simply the can in which the selected fragrance is held within. For the purpose of this prototype, a AXE men's body spray was selected. The bottle has a unique twist feature that ensures that no fragrance is administered while 'locked'. This was the primary reason for selecting this can as UNB-Fredericton was the primary grounds for testing and while indoors and near entrances we wanted to comply with the scent free policy. This locking mechanism on the bottle ensured no unplanned fragrance releases occurred.

Output

The output of the entire system is categorized into three senses: auditory, visual and olfactory. The audio and visual senses originate from the display of our Macbook PRO screen. Once the start button is selected, (unless the pause button is presse) the video will play until it's last frame. The olfactory sense is provided by the aerosol can experience the force produced from the spraying mechanism. This will occur during the 5 second window and only after the delay counter ceased.

System State Diagram

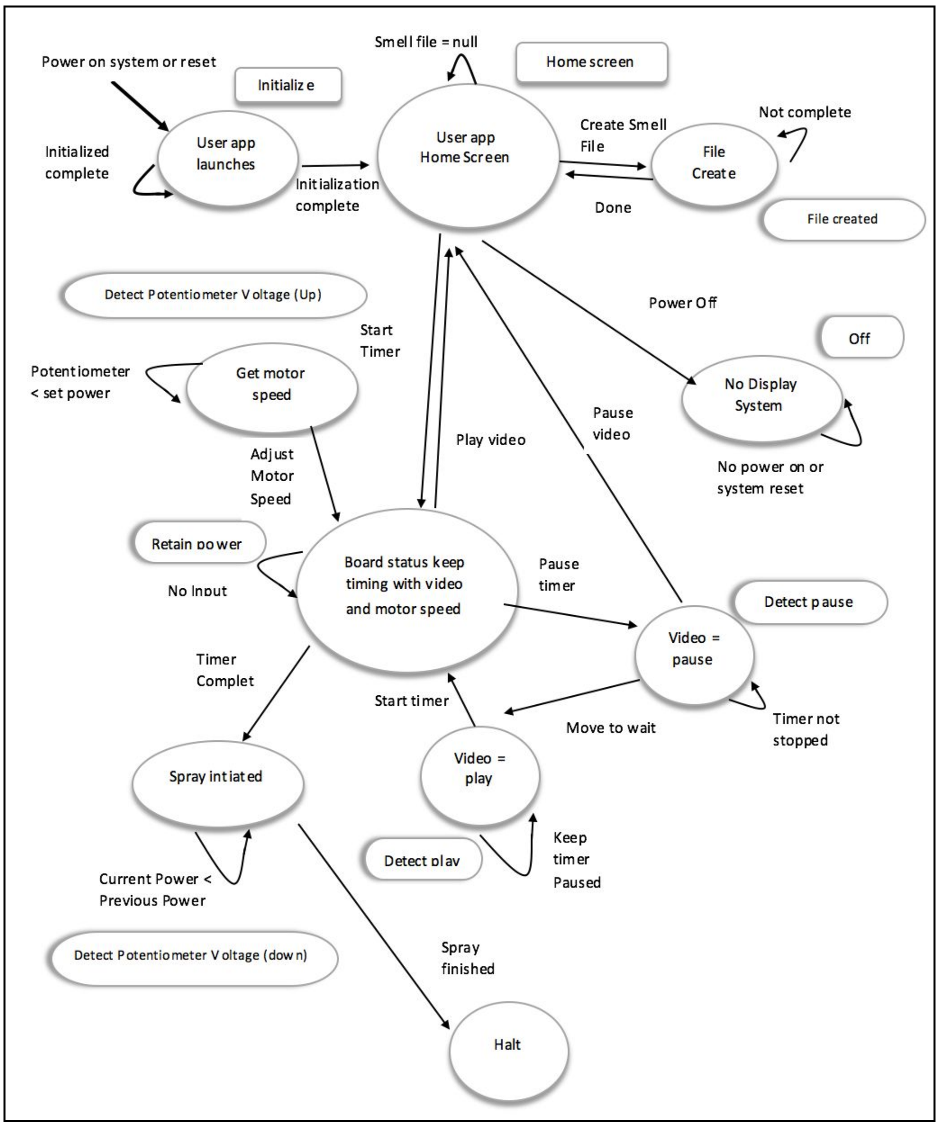

Figure 2: System State Diagram

The system state diagram above summarizes the individual states that the system will be in throughout operation. This diagram compliments the proposed principal of operation section and will be further discussed in the user manual section found below.

System Design

The purpose of the 'System Design' section is to compliment the section above and provide a lower level and detail-oriented description of both the hardware and software components that were designed and implemented into the prototype.

Hardware Design

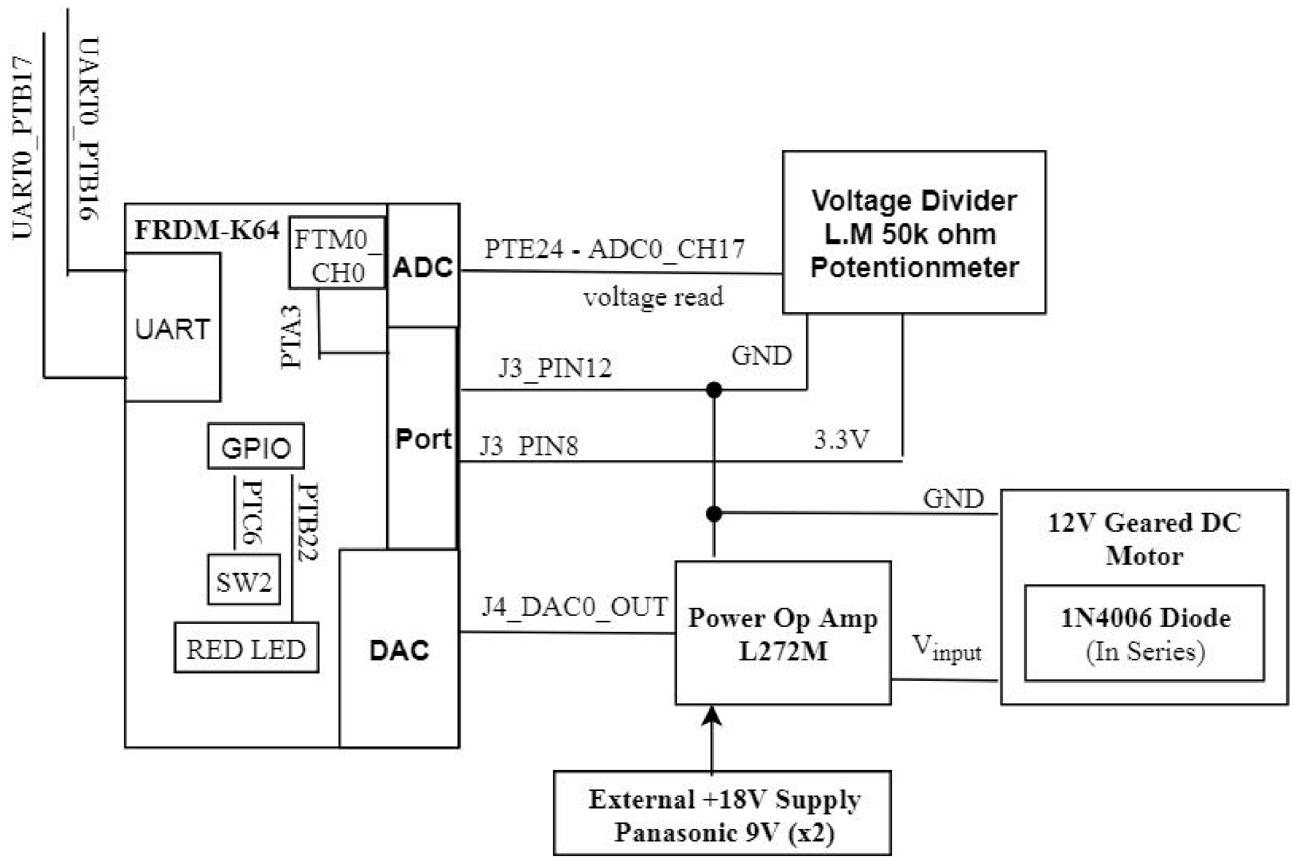

Figure 3: Hardware Design of prototype. Note UART0 is connected to the Macbook with the Java application via a USB C cable. This connection powers the FRDM-K64 board at a constant 3.3V. As well the DC motor is connected to the motor mount which is contributes to the spraying mechanism outlined in figure 1.

L.M 50 kΩ Potentiometer Voltage Divider

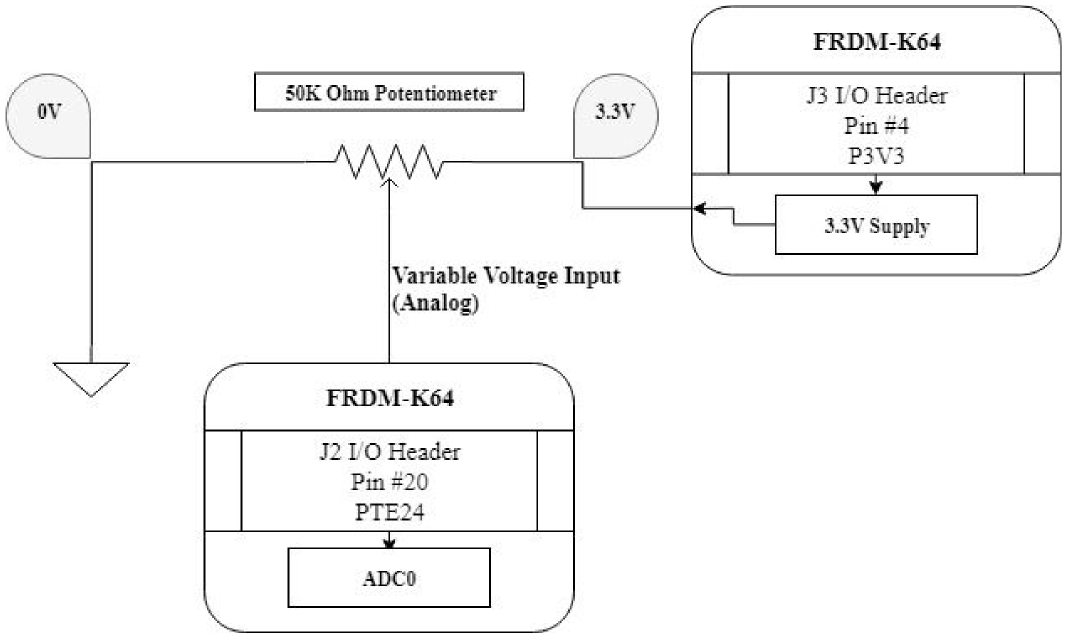

Figure 4: Breakdown of Analog Input to the FRDM-K64

Figure #2 illustrates the design of the 'adjustable knob' that allows the user to determine the amount of fragrance that will be dispersed over the 5 second window. This is accomplished by utilizing a 50 kΩ potentiometer (2). The first pin of the potentiometer is connected to ground that is provided by the J3 I/O header pm PIN12. Note this pin is utilized as the common ground for the entire circuit, this was done by extending the ground into a small breadboard. The third pin of the potentiometer is connected to the J3 I/O header on PIN8. This provides a 3.3V supply to the potentiometer. The middle pin of the potentiometer is connected to the J2 I/O header on PIN20. PIN20 interfaces with Port E PIN24 and is thus the external connection to our Analog-to-Digital Converter (ADC).

This component allows the user to simply twist the potentiometer to the desired input. Initially the potentiometer will be fully turned counter-clockwise, this will be the '0' input where 0 sprays will be administered from the can. As the potentiometer is turned clockwise, this will increase the amount of fragrances sprayed.

Power Amplification

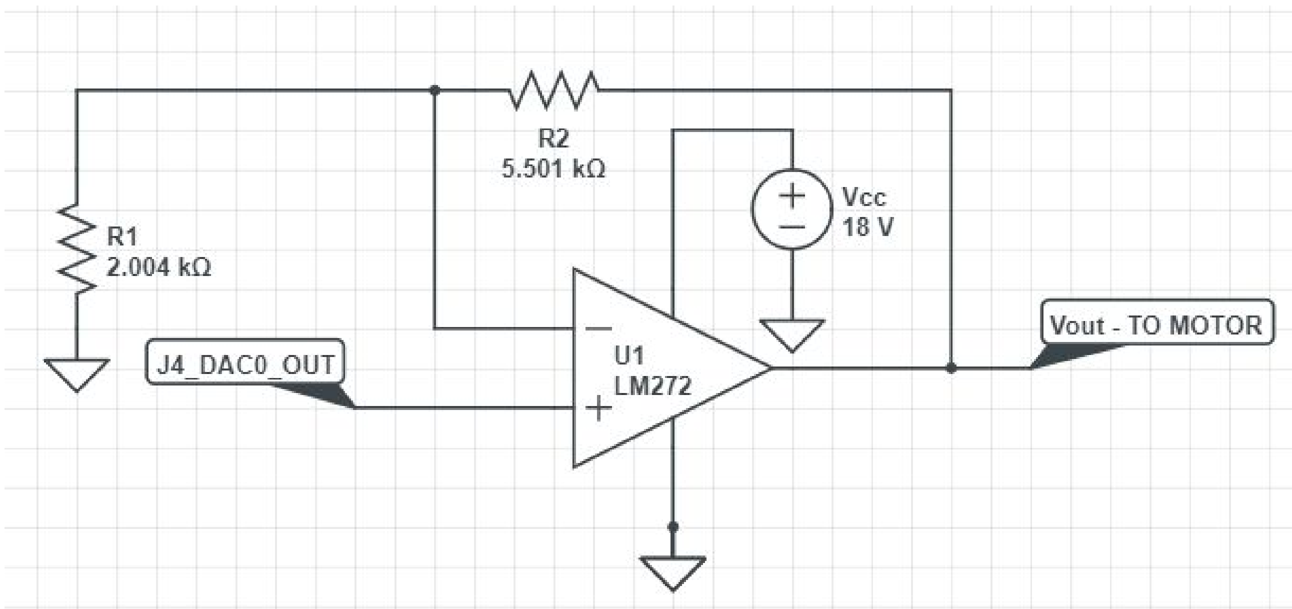

Figure 5: Power Amplification Circuit.

Power amplification was needed as the output of the DAC - J3_PIN11 only reached a maximum of roughly 3.29V. The geared DC motor used required a maximum of 12V and 65mA to attain 60 RPM. Note that 60 RPM was our desired maximum DC motor output as this would allow 5 'sprays' to be administered within the 5 second window. This was accomplished using a 56 kΩ E24 resistor, 2 kΩ E24 Series resistor, L272M power op amp (1) and two Panasonic 9V/1A batteries (connected in series). Figure 6 displays the approach taken to step up the voltage and current output from the board to achieve the maximum and minimum drive. The amplification circuit above creates the following transfer function:

$$\frac{V\,out}{V\,in} = 1 + \frac{R2}{R1}$$

With the nominal resistor values of R2 = 56000Ω and R1 = 2000Ω, we would achieve a gain of 3.6 V/V. This would allow a theoretical operating range of 0V-12.54V. However using a digital multimeter the resistance of the two passive components were reported to be R1 = 2004Ω and R2 = 5501Ω. This created a gain of 3.74 V/V and thus an operating range of 0V-12.28V. This operating range was perfect as during initial testing of the geared motor, it was capable of handle up to 15V without any signs of overheating or malfunction.

Geared DC Motor

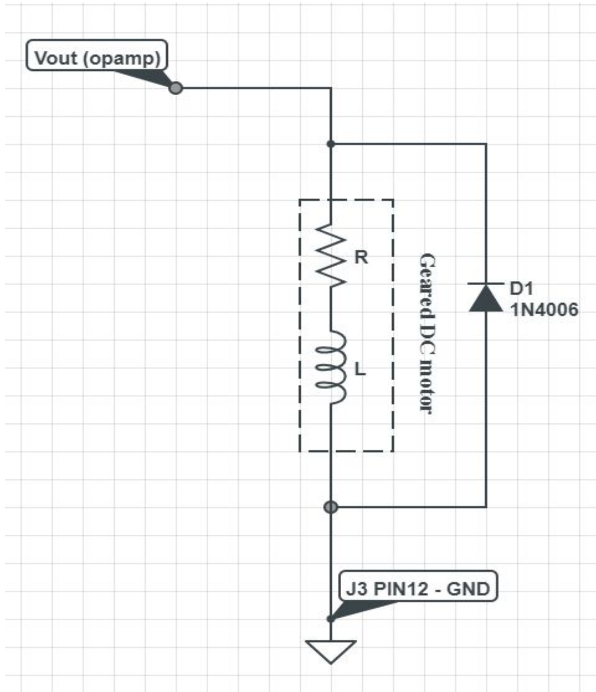

Figure 6: Implementation of the DC Geared Motor

The output of the L272M power op amp is connected to the positive terminal of the DC geared motor. In series to the DC motor is a 1N4006 diode. This diode was implemented based on the recommendation of the technicians found on D floor of Head Hall (UNBF). The purpose of this diode is to prevent unwanted discharge of current to flow back through the previous circuitry (i.e. the op amp). This is caused by the inductive qualities of the DC motor and can be summarized by:

$$V = L * \frac{di(t)}{dt}$$

Where the voltage across an inductive is indicated by the change in current multiplied by the inductance of the load. When V = 0V, we see that the change of current will become zero, and hence the current will be constant. This current will then seek a load in which it can be dissipated (i.e. the op amp) and thus the diode prevents such measures and ensures that the current ‘wheels’ around the loop seen above in figure 7. Hence the 1N4006 may be referred to as a ‘freewheeling’ diode.

Spraying Mechanism

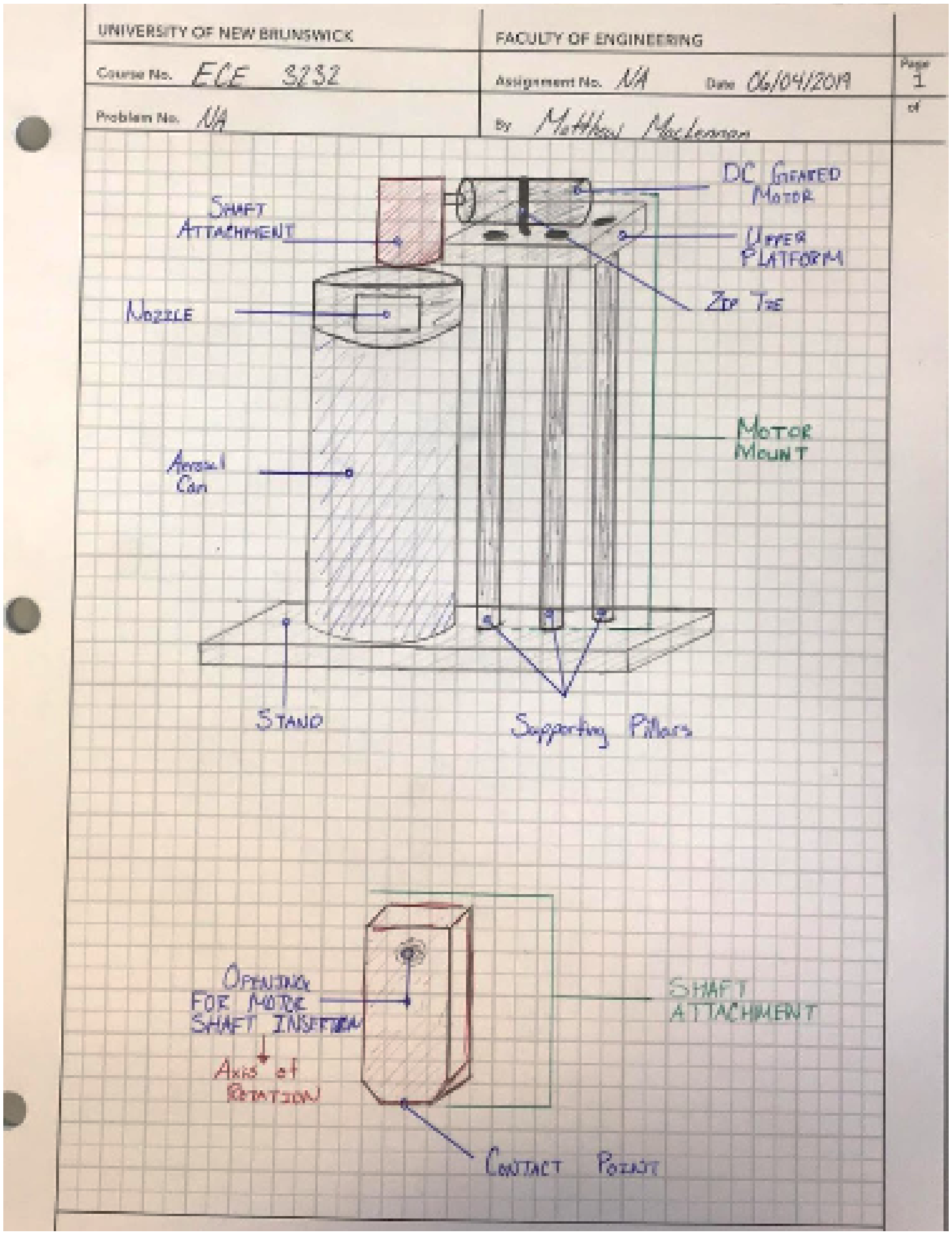

Figure 7: Sketch of the final iteration of the Spraying Mechanism

The spraying mechanism illustrated in figure 8 is representative of the final iteration that was implemented into the functional prototype. Previous iterations can be found in Dv2. The major changes between the previous and final iteration is the removal of the springs and pressure plate. The springs did not allow the design team to achieve the ‘critically stable’ point where only a small amount of force was required. This was due to the fact that the pressure plate was not fixed on the top of the aerosol can. Each time contact was made between the shaft attachment and nozzle of the can, the pressure plate was simply displaced do to the external forces. This caused no fragrances to be released.

The final iteration seen above illustrates that the aerosol can requires no pressure plate or added force (from the springs) in order for the dispense button to experience enough pressure. The chosen DC geared motor provides enough torque on it's own to supply the needed force in order for fragrance to be released. This greatly simplified the overall spraying mechanism when compared to the previous iteration. Due to this final advancement, the spraying mechanism is entirely operational.

Software Design

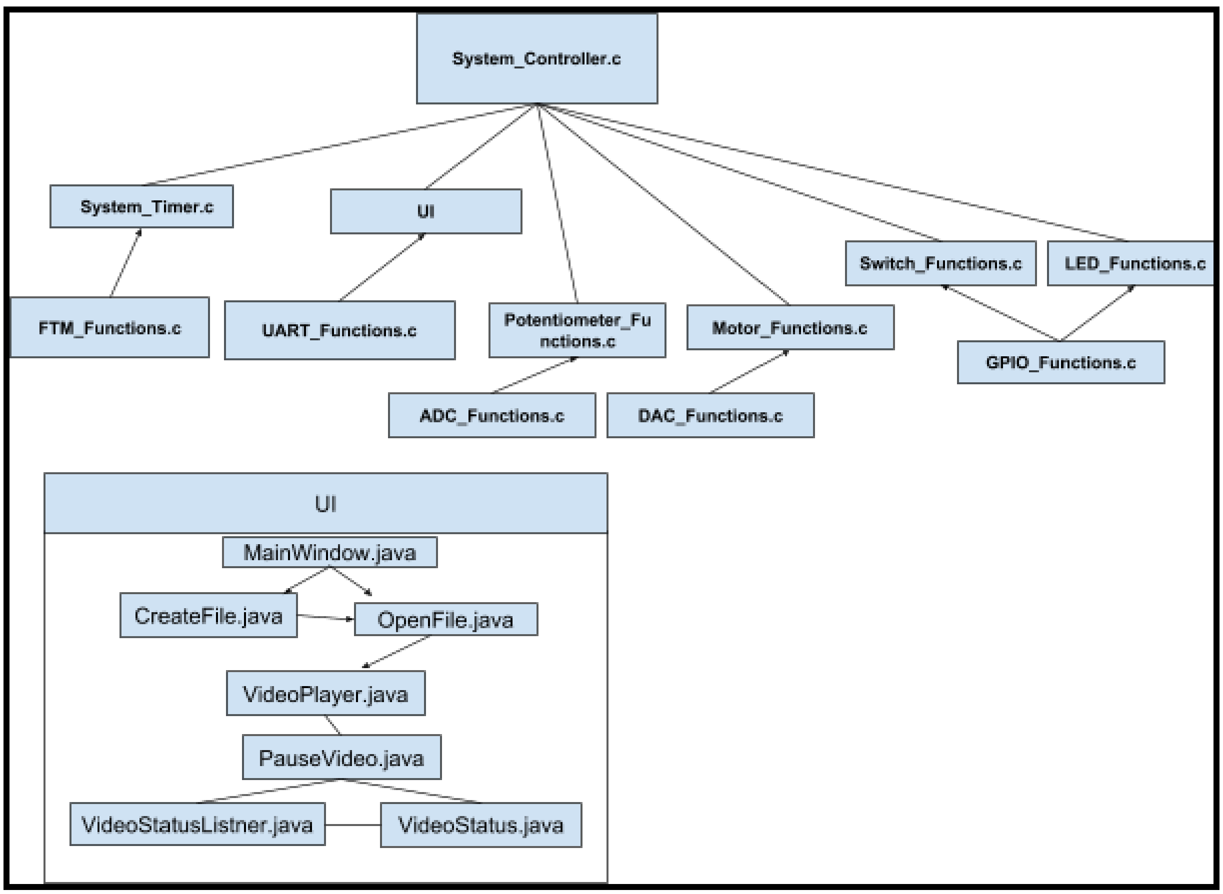

Figure 8: Software Breakdown for our application. All modules will be able to use the submodules connected beneath them. A more descriptive software architecture is described in the following sections.

Hardware Interface

Processing Information from Potentiometer

The speed of the motor is directly related to the voltage input read by ADC0_SE17. We call the sample function when needed, reducing the processing power required from the board.

Processing timing

The FlexTimer module (FTM) has been implemented to create a timed delay which must maintain time with the video displayed to the user from the G.U.I. This is accomplished by sending an interrupt signal ,activated by switch 2 on the GPIO module, to the G.U.I through the UART connection. This ensures that the video-time and timer stop and start together in unison preserving the system time.

```
/*Created By: Matthew MacLennan
* Description: Create delay to wait t seconds
* Input Parameters: t is time in seconds and PULSE_LENGTH is used to manipulate delay
time
* Return: None
*/
void delay(int t, int PULSE_LENGTH);
```

Displaying timer activated

The onboard LED was used for debugging purposes which is connected to the GPIO module on port B pin 22

FRDM-K64 Interface

The architecture used on the board follows a Modularized Software Architecture which is comprised of modules. This ensures that test cases can be reused as more software complexity is introduced.

Main.c

The Main module is the top level entity of the software architecture to interface with the FRDM. Its purpose is to communicate with the G.U.I and M.U.I to ensure user input is interacting as expected to produce a seamless experience between the video and spraying mechanism. It performs this by using the following methods.

```
/* Created By: Joshua Bell
```

* Description: Monitors the System and ensures that the spray is activated at the right time along   * with the right motor speed.
* Input: None
* Returns: None
*/
main - Main function when program starts

/* Created By: Matt MacLennan
* Description: Initialize all the Modules used
* Input: None
* Returns: None
*/
Initialize - Call all the initialize methods

/* Created by: Matthew MacLennan
* Description: Function to initialize ADC0 Channel 17 and PORTE for
* reading ADC data.
* Input Parameters: None
* Return: None
*/
void init_ADC();
/*
 * Create By: Joshua Bell
 * Description: Function to initialize DAC0 module
 * Parameters: None
 * Returns: None
 */
void init_DAC();
/*Created By: Matthew MacLennan
* Description: Initializes the FTM module for FTM0_CH0 on PTC1 (header J1_5).
* Configures FTM0 for rising edge input capture.
* Input Paramaters: None
* Return: None
*/
void init_FTM();
/*
 * Created By: Joshua Bell
 * Description: Function to initialize GPIO modules that are used for the switches
 * Parameters: None

```
 * Returns: None
 */
void init_GPIO();
```

System_Timer.c

```
/*
 * Created By: Joshua Bell
 * Description: Function convert UART communication time from type char to int
 * Parameters: a is a number between 0-9, count keeps track of largest base 10 number,
current number is the return number
 * Returns: Integer time in seconds
 */
int lookUpTable(char a,int count, int currentNum);
/*
 * Created By: Matthew MacLeannan
 * Description: Function waits for a time to spray input
 * Parameters: None
 * Returns: Integer time to release spray
 */
int getSprayTime();
```

LED_Functions.c

```
/*
 * Created By: Joshua Bell
 * Description: Turn the red LED on the board
 * Parameters: None
 * Returns: None
 */
void LED_on();

/*
 * Created By: Joshua Bell
 * Description: Turn the red LED off
 * Parameters: None
 * Returns: None
 */
void LED_off();
```

Potentiometer_Functions.c

```
/*
 * Created By: Matt MacLennan
 * Description: get the voltage input and determine the speed needed input for motor speed
 * Parameters: None
 * Returns: Integer that can be inserted into setMotorSpeed
 */
int getMotorSpeed();
```

Motor_Functions.c

```
/*
 * Created By: Joshua Bell
 * Description: Set motor speed
 * Parameters: Integer that setsMotorSpeed
 * Returns: None
 */
Void setMotorSpeed(int Speed);
```

Switch_Functions.c

```
/*
 * Created By: Joshua Bell
 * Description: Function intialize SW2 for use on the FRDMK64 board
 * Parameters: None
 * Returns: None
 */
void switch_init();

/*
 * Created By: Joshua Bell
 * Description: Function wait until switch has been pressed
 * Parameters: None
 * Returns: int witch indicated a boolean value
 */
int switch_pressed();
```

Java Interface

The java interface is developed with a Model View Controller architecture (**MVC**) . This decouples the G.U.I from the data being exchanged between the board and the java application.

MainWindow.java

This is the home window where the user can either choose to create a smellOvision file from aMP4 file or selected an existing smellOvision file to play.

CreateFile.java

This window allows the user to select a MP4 file and convert it into a smellOvision file where the user must input a time in which to activate the spray. After a file has been created the user will be redirected to the OpenFile.java window.

OpenFile.java

Here the user can select a smellOvision file to be played through the application. Once a file has been selected it routes the user to the VideoPlayer.java .

VideoPlayer.java

VideoPlayer.java displays a window where the user can press the start button to begin viewing the video.

PauseVideo.java, VideoStatusListener.java, VideoStatus.java

All three of these java classes are used to monitor the status of the video being played and update the G.U.I if the a pause/play has been detected from the board.

Term Test Plans

The test plans outlined below took place over the Winter Academic Term for UNB Fredericton. The design group was formed on January 25th (note that some test plans are identical to those performed during labs as identical components were being tested). Since then, the team has performed numerous test plans on each sub component outlined in the sections above. Below is a table that serves to summarize the results of each unique test plan. Debugging was primarily accomplished by utilizing the built-in debugger found in the Kinetis IDE. This debugging approached heavily relied on utilizing the step over/step into feature, reading registers and inserting breakpoints at areas of interest.

Table 1: Summary of Test Plans

Component(s) Tested		
Test Number	**Sub-Module**	**Results**
Description of Test (i.e. its purpose)		
GPIO		
1	Red LED - PTB22	Success
Turn on an LED by hard coding it's value to 0. (Active Low)		
2	Red LED - PTB22	Success
Turn on an LED by using the PSOR, PTOR and PSOR registers to toggle it's value to ensure it is 0.		
3	Red LED - PTB22 + SW2-PTC6	Success
Turn an LED on and off via SW2. When the button is pressed, the LED should come on		
ADC		
4	PTE24-ADC0	Success
Test the ADC by giving simple inputs. Values given were: 0V & 3.3V (supplied by the board) and 1.78V via voltage divider		
5	PTE24 and L.M Potentiometer	Success
Connect the ADC to potentiometer obtained from the maker space. A voltmeter was used to accurately read the input into port E-pin 24 and this value was compared to the value found in the test register once it was converted from hex to decimal. Many values were testign ranging from 0.001V - 3.297V.		
UART		
6	UART0_PTB(16&17) + PuTTy	Success
Test the transmission and receiving features of the UART capabilities of the board. PuTTy was used on a desktop to verify. Both characters and numbers were sent to and displayed on the PuTTy terminal and values were sent from user input via PuTTY and stored within the a register found on the board.		

7	UART0_PTB(16&17) + Java App.	Success
By launching Eclipse development environment and implementing the important library necessary for serial communication we were able to test UART connection by outputting the status of the connection by invoking `serialPort.openPort()` . This displays whether the connection was successful or not in the eclipse console.		
FTM		
8	FTM0_CH0 &PTA3 + RED LED	Success
Using the FTM module, integer long delays were implemented. The length of which the delay took place was obtained from the user upon each cycle. An LED was lit at the beginning of the program and would turn off after the timer delay reached zero.		
9	FTM0, RED LED and SW2	Success
Test the software to determine whether the FTM is capable of temporarily halting counting. The red LED was lit while the FTM overflow bits were counted and the LED turned off when SW2 was pressed to show that the overflow bits were not being counted (i.e. timer halted).		
DAC		
10	DAC0_OUT	Success
Hardcoded values were given to a register and then written to the output register of the DAC. A voltmeter was used to verify that the outgoing voltages from the board matched the hard coded value.		
11	DAC, L.M Potentiometer and ADC	Success
The potentiometer with a 3.3V supply provided by the board was used as an input signal to ADC0. The data read in by the ADC was then written to the DAC and a voltmeter was used to verify the output. This proved that the both functions worked together and would allow the group to meet analog in/out specifications.		
Power Op Amp		
12	L272M, E24 Series Resistor and UNB electronics breadboard	Success

Testing the viability of the power amplifier. Negative feedback was used and configured to provide a gain of 3.75V. The 3.3V supplied by the board was the input signal and voltage was read at roughly 12.2V.		
13	L272M, E24 Series Resistor and UNB electronics breadboard, L.M Potentiometer	Success
The same set-up for test 12 was used, however a small addition was made. The potentiometer was added at the start to allow variable input. This ensured that the op amp could handle the 0-3.3V output voltage made by the DAC and scale it appropriately.		
DC Geared Motor		
14	DC Geared Motor, 1N4006 and components listed in test 13.	Success
The DC geared motor was connected as a load to the output of the amplifier. The freewheeling diode was connected in parallel to the motor to ensure the amplification circuit was protected.		
Java Application		
15	smellOvision file is created	Success
To test this, an existing MP4 file on the PC running the java application is needed. By following the create a smellOvision file steps in the user manual ensure that the smellOvision file is actually created and stored in the location you selected during the creation.		
16	SmellOvision file is runnable	Success
Once you have successfully created a smell file continue this test by opening the file in the java application OpenFile window described in the User manual section. After clicking the start button ensure that the video plays as expected. If the video does not play then the test is failed.		
17	pause/play works	Success
Begin playing a smellOvision file as described in test case 16. Once a video is running press switch 2 and make sure the video pauses. Press switch 2 again to resume playing. If this does not happen the test case is failed.		
Final Test (All previous Modules)		
18	All components outlined in figure 1	Success

All components were incorporated together to ensure that the entire prototype is functional. The G.U.I was opened, with the MP4 uploaded, smell file created and delay time inputted. The potentiometer was set completely clockwise,counterclockwise and multiple areas in between. The video was played and the pause/play feature (SW2) was checked while also verifying the the spraying mechanism was functional.

Evaluation

Acceptance Tests

The Acceptance tests were highlighted in a section at the beginning of the document, below is a table which summarizes the various tests and the performance of our prototype.

Table 2: Summary of Acceptance Tests

Test Number	Criteria	Outcome
1	Acquire user input via the functional M.U.I and G.U.I.	The user is able to select the delay time, video to be played, amount of fragrance dispersed all through the G.U.I and M.U.I.
2	Proper video playback without any hindrances as listed in the System Characteristics and Constraints section.	Video is played at original quality with sound. No buffering was encountered.
3	Functional pause/play button.	The play/pause button is capable of stopping both the timer as well as the video being displayed on the PC
4	Appropriate window over which the selected scent is administered.	A five second window was hard coded at which fragrance may be released.
5	Appropriate amount of fragrance is released.	Various potentiometer inputs were used. 0-5 sprays were given within all test conditions. Although the shaft mount must be repositioned to have highest

		accuracy.
6	The entire procedure is straightforward for the viewer.	The user manual found in the section below outlines the simplicity of our design. Asked 2 random UNB students to review the document and approved that it was straightforward.

Current Status

As of April 6th 2019, the entire design outlined above has been built and tested. The prototype is turned on when the user plugs in the USB C cable to board and PC, it will be on ‘standby’ until further commands are given. The user may open the Java Application launching the G.U.I where a MP4 file is uploaded, converted to a ‘smell file’, the desired delay is inputted and finally the G.U.I prompts the viewer to start the video.

Prior to pressing play, it is recommended that the viewer adjusts the potentiometer found above the USB C port for the FRDM-K64 board to select the desired number of 0.25second sprays of the aerosol can. The minimum number of sprays is 0. The maximum number of sprays in 5. The spray(s) (if any) will take place once the timer stops and occur within the 5 second window.

Once the start video option is selected on the G.U.I, the user may comfortably watch the video and has the option to hit the pause button (on the FRDM board) to stop the video if needed. The user may hit the same resume the video by pressing the same button a second time.

The FRDM-K64 board, amplification circuit, Panasonic batteries and spraying mechanism are all housed within a large box. This has the appropriate cutouts where the user may access the USB C port, SW2, potentiometer and the cutout at which the fragrance will be released through. Conveniently, the container is large enough to act as a stand for the laptop being used and is extremely light on it’s own.

Future Work

As summarized above, the prototype is functional and is capable of accomplishing the problem statement as defined at the beginning of this document;however, our design team feels that improvements could be made to increase the viewers experience. They are found in the following list below:

1. Designing and implementing a PCB that could be mounted as a shield onto the FRDM board will save space as well as further protection and other added benefits could be easily

incorporated. This would also ease demands in mass manufacturing where currently all circuits are created manually using a small breadboard.
2. The casing size could be optimized where it is lease cumbersome to carry.
3. Staining the external wooden surface of the case to increase it's visual appeal.
4. Appropriate labelling is needed on the external surface of the case.
5. Allowing multiple delays to be inputted into the G.U.I. This would enable the viewer to decide the number of 5 second windows that occur throughout the playing of the video.
    a. Additionally, the user would be able to define the amount of fragrance released at each event (where a single event corresponds to a inputted delay)
6. Allowing some storage inside the container. This would allow the viewer to interchange the selected aerosol can.
7. Continuing work on the G.U.I, such as:
    a. Improving the physical display that resembles that of a commercialized program
    b. Allowing multiple files to be uploaded and stored on a cloud
8. Implementing a wireless controller instead of the push button. This could either be a physical remote or making the board compatible with voice commands.

User Manual

Smell-O-vision is a product which allows users to combine video and smell into one enjoyable experience.

Set-up:

1. Download the java application.
    a. This can be done by contacting jbell5@unb.ca or mmaclen2@unb.ca, please indicate your name and that you are asking for the smell-O-vision java application runnable jar.
2. Acquire an assembled smell-O-vision product(Batteries included).

3. Launch the smellOvision.jar. It will present the user with the following window.

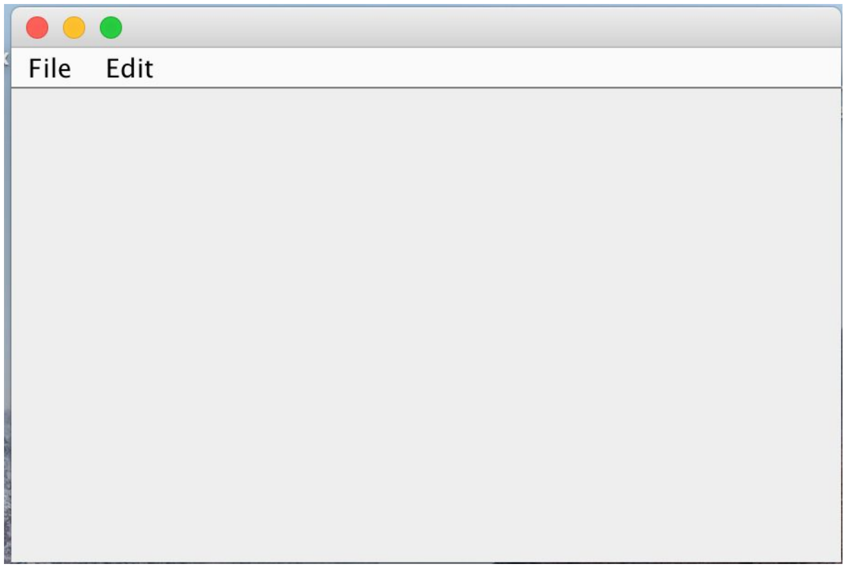

Figure 8: This is the Home screen window. From here the user can select the following options.

4. If you do not already have a file with the .smellOvision file extension you will need to create one. Select the **Create New…** option.

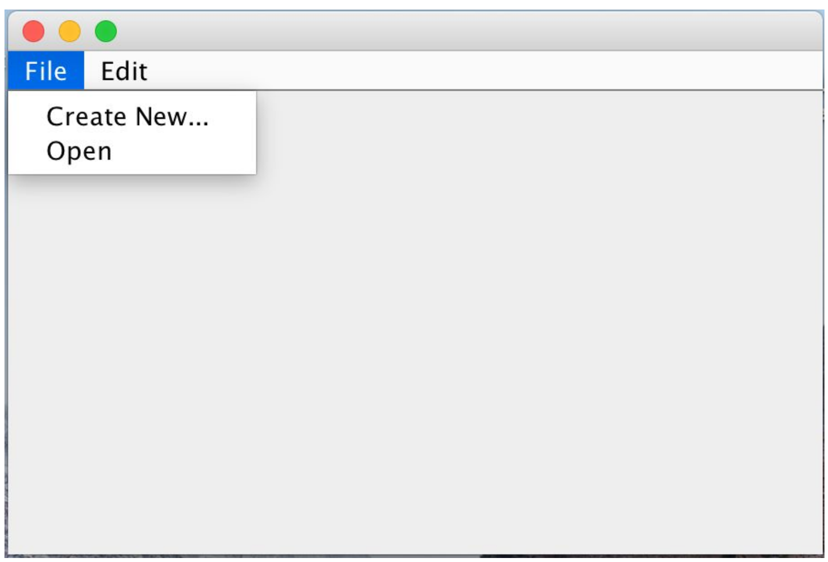

Figure 9: User options view.

5. Now that you have selected to create a new file you will be brought to the createFile window. The open file button only accepts files with the .MP4 file extension for compatible conversion to smellOvision file.

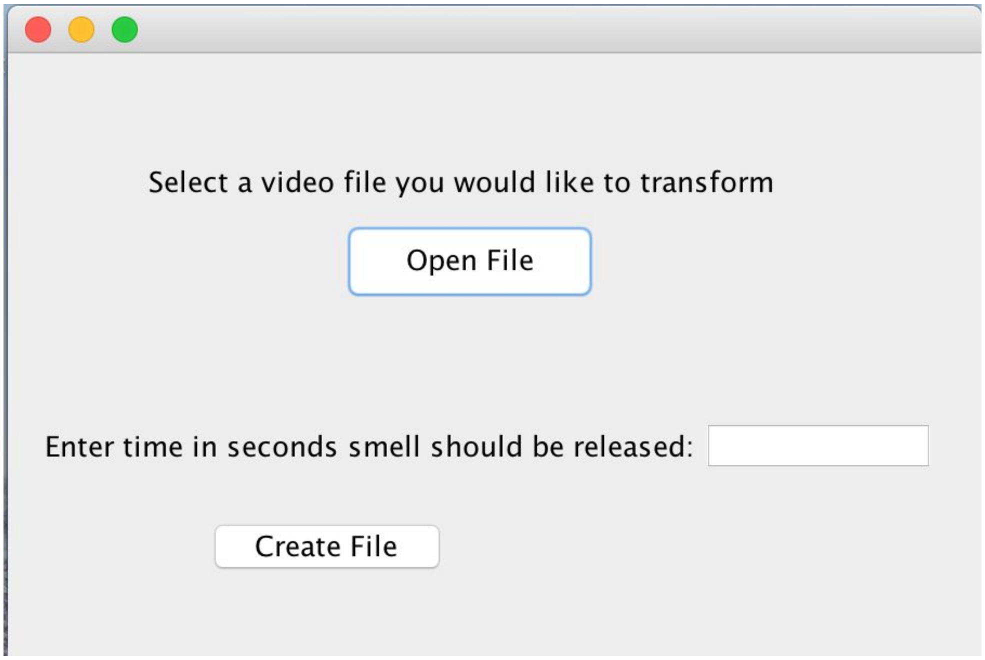

Figure 10: Create smellOvision file window

6. Once you have created a smellOvision file or if you already have one you can access this by selecting the **Open** option from the user options view. Continue through the file explorer by selecting the smellOvision file you wish to consume.

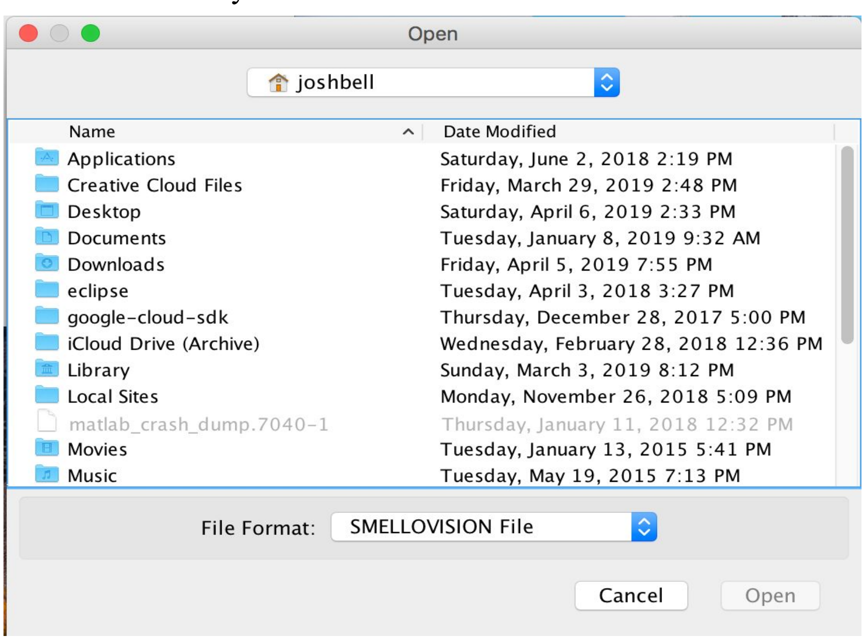

Figure 11: Open smellOvision window.

7. Before selecting the start button presented on Figure 12. Make sure that you have connected your smellOvison product to your PC through the USB C, which you are running the smellOvision.jar on.

8. Next it is suggested setting the potentiometer located on the left hand side of the smell-O-vision casing all the counter-clockwise so that the spray volume is minimized and increasing the potentiometer until desired setting is reached.

9. Now that everything is ready press start and enjoy watching your smellOvision experience.

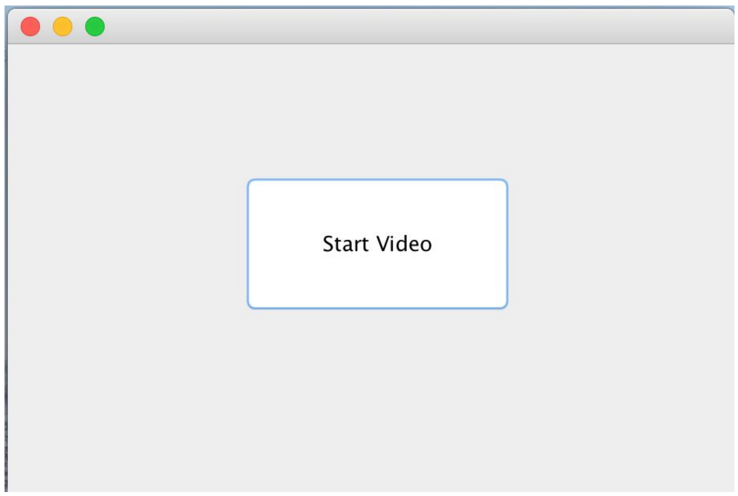

Figure 12: The VideoPlayer window

10. If at any point during the video the user wishes to pause the video he/she may do so by pressing switch 2 located on the left hand side indicated by figure 13.

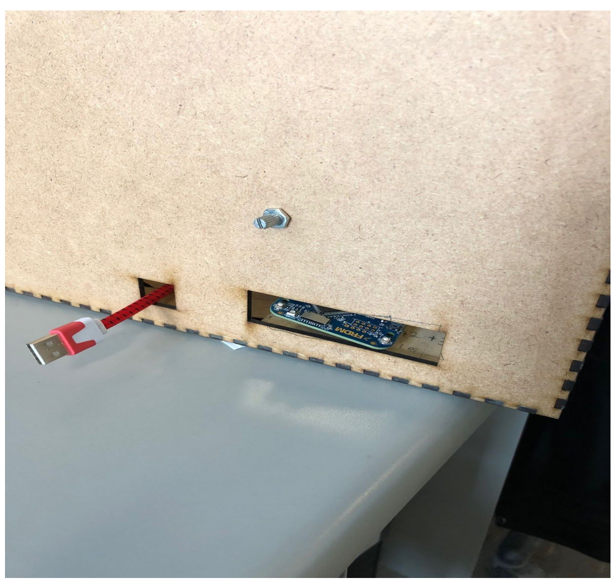

Figure 13: SmellOvision Mechanical user interface

References

[1] ON Semiconductor L272/L272A Dual Power Operational Amplifier,
https://www.onsemi.com/pub/Collateral/L272A-D.pdf

[2] Bourns Pro Audio, PDB18 Series - 17mm Rotary Potentiometer,
https://www.bourns.com/docs/Product-Datasheets/PDB18.pdf?fbclid=IwAR2Q2yDVAsALgaJgOi6S6VGFI6guWnVSOGn2yAOJqJL2T-gm8qjWr6-sRZQ

[3] Diodes Incorporated, 1N4001-1N4007 1.0 Rectifier Series,
https://www.diodes.com/assets/Datasheets/ds28002.pdf?fbclid=IwAR1IVUXTrrXQCGStxIFaM5KBPS0zkvPfSWwXKP6nxXIK5fUQpIGD6GnygsE