Worcester Polytechnic Institute

# RBE 2002 Final Project Report

## Romi and Juliet

Team 14

Michael Beskid, Lily Durkin, Greg Marshall

Equal Division of Labor

Unified Robotics II: Sensing

May 13, 2021

# Contents

Introduction ..... 3
Overview ..... 3
Fight Scene ..... 3
Balcony Scene ..... 4
Final Scene ..... 5
Solutions and Justifications ..... 7
Fight Scene ..... 7
Balcony Scene ..... 10
Final Scene ..... 13
Performance ..... 15
Fight Scene ..... 15
Balcony Scene ..... 15
Final Scene ..... 16

# Introduction

For the RBE 2002 Final Project, our team applied our knowledge of robotics to design, build, and program robots to perform three pivotal scenes from William Shakespeare's famous work Romeo and Juliet. Specifically, our adaptation of the production entitled "Romi and Juliet," made use of three Pololu Romi robot kits to enact the fight scene, balcony scene, and final death scene from the original work. The robots were modified and outfitted with several sensors in order to navigate the stage and interact with another while performing each scene fully autonomously.

# Overview

This section contains a high-level look at each of the scenes which were recreated with our robots. A description of the modifications made to each robot including the sensors equipped is given, in addition to an overview of the methods used to implement the desired functionality.

## Fight Scene

The fight scene makes use of all three robots, playing the roles of Mercutio, Tybalt, and Romi. This scene begins with Mercutio and Tybalt circling each other about a central point while Romi watches from a distance away. Tybalt then turns to lunge and stab Mercutio, and Mercutio dies dramatically. Similarly, Romi then turns to lunge and stab Tybalt for revenge, before fleeing the scene as Tybalt dies. Mercutio is outfitted with an IR emitter LED and an IR receiver, as well as a red LED. Tybalt is configured with an IR receiver, IR flame detector, and red LED. A 3D printed bumper was attached to the front of the robot, and a mount was fixed to the side of the robot to hold an April Tag in front of the left wheel. Lastly, Romi is given an IR receiver, a similar 3D printed bumper, and an OpenMV camera attached to a front facing mount. Tybalt and Mercutio use kinematics in the beginning of the scene to drive in a circle of a specified radius and keep track of encoder counts to stop after two revolutions. IR communication from Mercutio to Tybalt and Romi tells them when to seek and ram their targets. Tybalt uses the IR flame detector for seeking Mercutio, while Romi uses the OpenMV camera to seek Tybalt. All three robots detect collisions with the onboard IMU. A picture of this scene is shown in Figure 1. Mercutio is on the left of the scene, Tybalt on the right, and Romi in front.

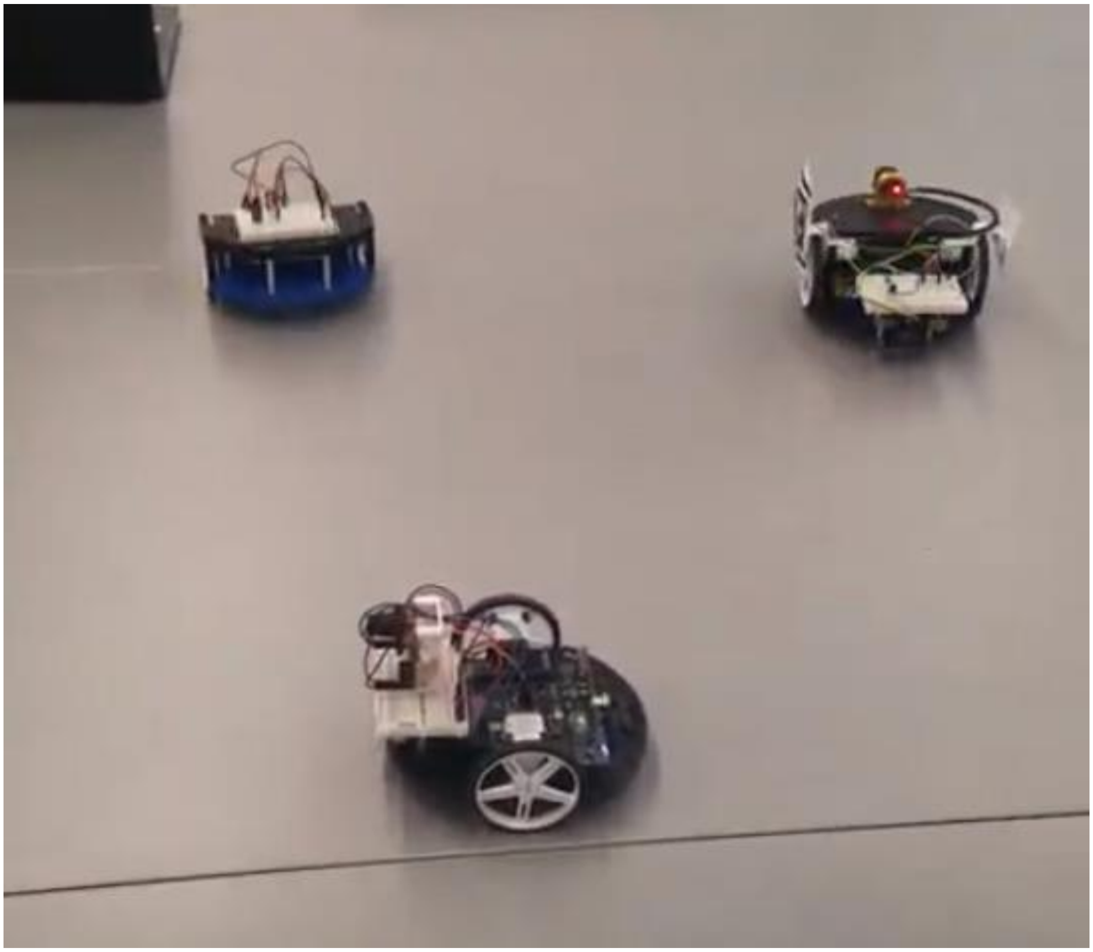

*Figure 1: Fight Scene*

## Balcony Scene

The balcony scene employs two of our robots acting as Romi and Juliet. In this scene, Juliet begins at the bottom of a ramp, and Romi begins a distance away, perpendicular to the ramp direction. Juliet drives up the ramp and stops at the top before signaling to Romi. Romi then drives up to the base of the ramp below Juliet and turns to face towards the bottom of the ramp. With Romi in position, Juliet drives back down with the ramp with Romi following alongside her, and the two continue driving a short distance before coming to a stop. For this scene, Juliet is equipped with an IR receiver, an IR emitter LED, a reflectance array sensor mounted to the front of the robot, and a mount to hold an April Tag over the left wheel. Romi has an IR receiver, a Sharp IR proximity sensor mounted to the front of the robot, and an openMV camera attached to a side facing mount. Juliet uses the IMU to detect the slope of the ramp at the beginning of the scene to drive up to the top and stop. IR communication between the robots is sued for Juliet to signal to Romi to come over. Romi then approaches the ramp and uses the IR proximity sensor to stop the correct distance away before turning 90 degrees with the help of the motor encoders. As Juliet drives down the ramp and continues beyond, Romi uses the OpenMV camera to track the April Tag mounted to Juliet and follow alongside her. Lastly, Juliet uses the reflectance array sensor to detect a tape line at the intended stopping point, and again uses IR communication to signal for Romi to stop. A picture of this scene is shown in Figure 2. Juliet is pictured at the top of the ramp and Romi is seen approaching the base of the ramp.

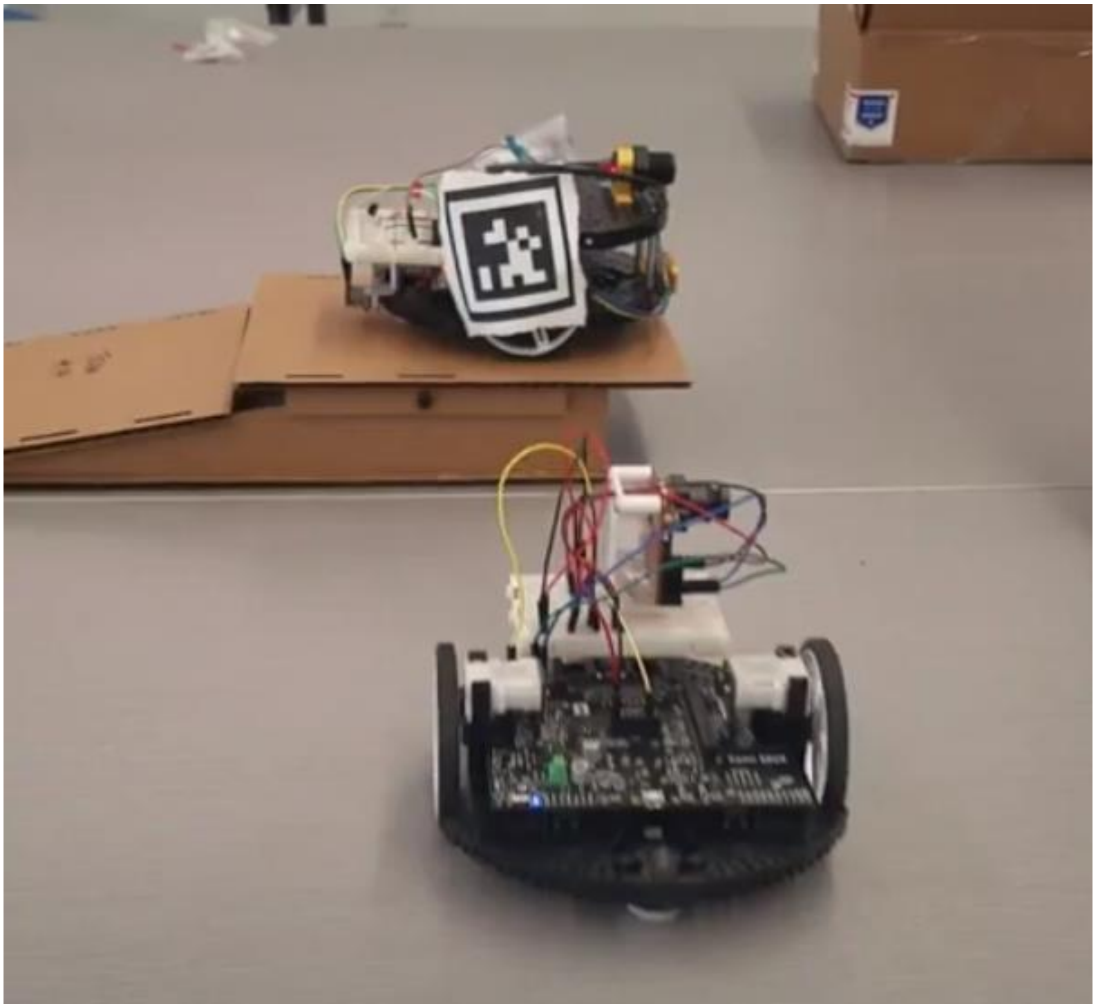

*Figure 2: Balcony Scene*

## Final Scene

Two robots are used in the final scene, again playing the roles of Romi and Juliet. This scene begins with Juliet positioned a small distance away from the back wall towards the right of the scene. Romi enters from the left, carrying a cup of poison, and drives towards Juliet before stopping to find her presumably dead. At this point, Romi tips the cup and takes the poison, drives up to give Juliet a final kiss, then backs away and slowly dies. Juliet awakens a moment later to find Romi dead. Juliet is simply outfitted with an IR receiver and a yellow LED for this scene. Romi is configured with an IR emitter LED, an IR receiver and red LED. An ultrasonic rangefinder is mounted to the side of the robot and a Sharp IR proximity sensor is mounted to the front. Additionally, a 3D printed poison cup and tipping mechanism are mounted to the back of the robot, which features a servo motor to actuate the mechanism. To begin the scene, Romi drives along the back wall while using the ultrasonic rangefinder for wall following. The Sharp IR proximity sensor is used to detect when Romi gets close to Juliet, then the servo position is set to tip the cup. Inverse kinematics is then used to close the remaining distance to Juliet. Romi next spins around with decaying speed as he dies, turning on the red LED as he stops and using the IR emitter to signal to Juliet. This signal is picked up by Juliet's IR receiver, causing the yellow LED to light and a small movement from Juliet to signify her awakening. A picture of this scene is shown in Figure 3. Romi is on the left and Juliet is on the right.

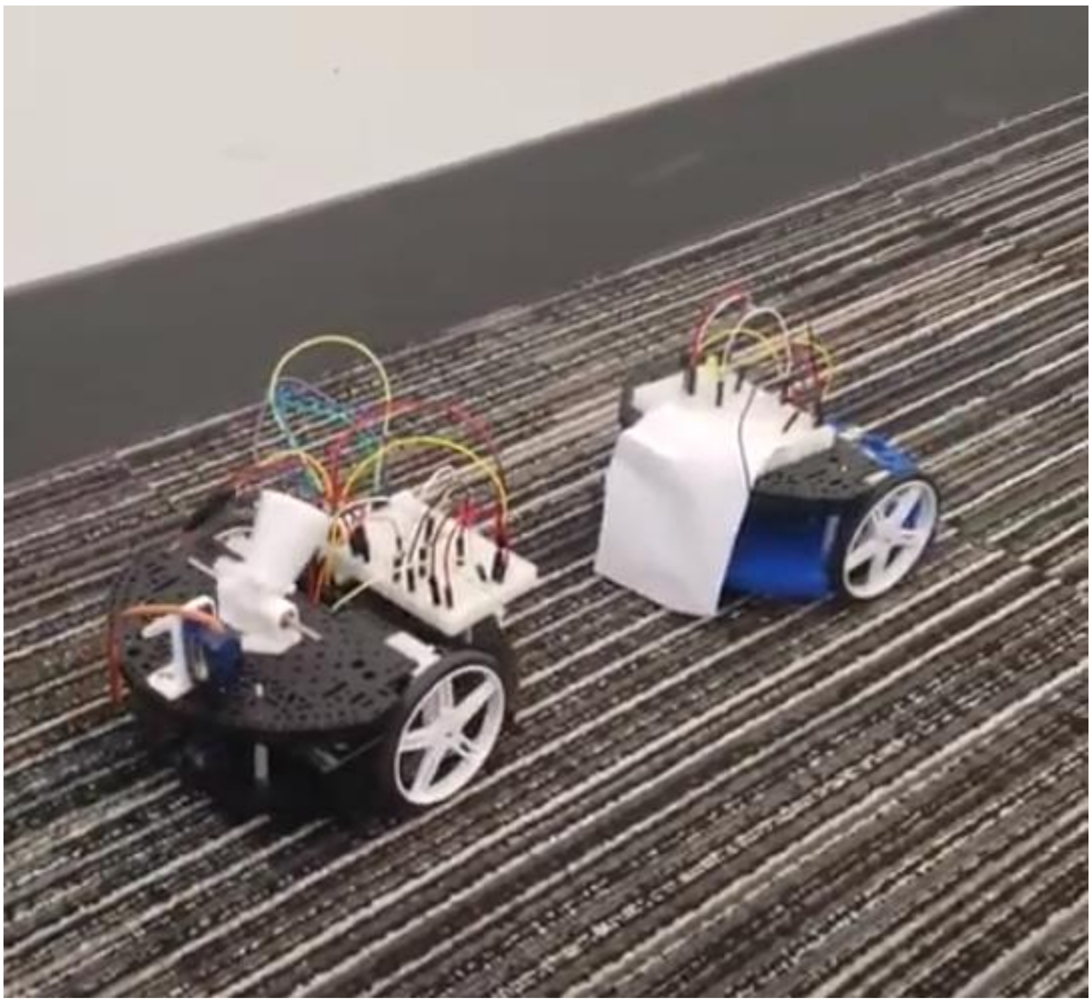

*Figure 3: Final Scene*

# Solutions and Justifications

## Fight Scene

A major challenge in the first scene is getting Mercutio and Tybalt to both drive in a circle about a point. This challenge has two major components, driving in a circle and correcting for the error that accrues as it drives in a circle. The first issue was handled using the equations derived from the forward kinematics equations for instantaneous center based prediction, see equation below. This equation worked by providing speed values for the left and right motors in cm/s based on a given radius and speed(U0) and by setting the speeds of the motors to these values it causes the robot to smoothly follow a circle of the specified radius. To help minimize the error accrued throughout the circling process PID control was utilized to make minor adjustments to the speeds of the two motors getting its error by calculating the difference between the set radius and the distance between the robots target point and current position. By using both PID control and the arcing equation the robot was able to circle with minimal error in its position, error being defined as the distance between where it is at the end of a revolution and where it started, accruing approximately 7cm of error after driving around a circle with a diameter of 100cm 10 times averaging under 1cm of error per revolution. In addition to driving in a circle the robots needed to know when to stop. This was achieved by checking how close the robot was to its starting position, incrementing a value on the rising edge of it going within a certain threshold of its starting position. One alternative that was considered was having a beacon in the center of the circle for both of them to track with an IR emitter for Tybalt to track and April tags for Mercutio to track, the downsides to this method are that it would add an obstruction to the field between potentially between Tybalt and Mercutio and would also require Romi to forgo seeking Tybalt barring a third method for seeking. Another possible alternative that was considered was to create a short wall that was sloped on the outside to allow the robots to wall-follow with, the downsides to this were that it would require an obstruction between Romi and Tybalt and a circle large enough and tall enough for them to follow due to the sensors limitations. The upside to these two methods is that they both utilize external stimulus meaning that they, if implemented properly, would be resistant to drift and human error in starting positions. Through testing though the IK based method described above proved to be viable due to the drift being fairly minimal and the human error introduced by robot placement was counteracted by our methods for seeking/ramming described below.

$$\mathit{Arcing\ Equation}$$
$$V_L = \frac{U_0}{2}\left(2 - \frac{B}{R}\right)$$
$$V_R = \frac{U_0}{2}\left(2 + \frac{B}{R}\right)$$

Another major challenge in the fight scene was getting the robot to ram into each other. Because the robots won't start facing their target, Tybalt because he will be tangent to the same circle as Mercutio and Romi because his IR receiver has to have good line of sight to be awoken, both robots have to turn to face their targets. On both robots this was achieved using cameras, the IR camera (flame detector) on Tybalt and the OpenMV camera on Romi. By calculating the current offset from the center of the camera, in the x axis, for both cameras then normalizing this value to a -1 to 1 the percent offset of the target, the IR emitter on Mercutio for Tybalt and an April tag on Mercutio for Romi, which could then be fed to a PID loop to control the angular velocity of the robots. Additionally, by normalizing both the cameras outputs to the same range little to no tuning was required after tuning the first robots PID loop. This Seeking state was exited upon the offset falling below a certain threshold entering the second part of this challenge ramming. Because in the previous state the there is a threshold around the target there was guaranteed to be error when the robot started driving which would compound with the error accrued on the trip to the other robot. Because of this if the robots drove straight forward the error could cause the robot's attack to be a glancing blow rather than a harsh strike possibly preventing the IMU from detecting the collision. To minimize this error a PID control loop was used taking the same error as the Seeking step but instead of making large changes to the angular velocity it made small adjustments to the target speeds of the motors, in effect slightly altering the angular velocity to maintain a proper heading.

During the scene the three robots need to communicate amongst each other as well as detect other external events to control change between different states. During their ramming state the robots needed to detect the collision, that is both the robot that is doing the ramming and the one being rammed into need to know that it had happened. To achieve this we utilized an algorithm similar to the one implemented on the Romi's IMU by detecting a rising or falling edge with a amplitude above a set threshold, in the accelerometers xy plane storing the time and which direction the edge was in. Then when an opposite edge again with a magnitude over the set threshold is detected, within a set time threshold of the initial edge, a flag is set in the IMU class. This flag is then left true until it is requested with a function where it is returned and its value is reset. In addition to detecting collisions with the IMU Romi needed to detect Mercutio's death. This was accomplished by equipping Mercutio with an IR emitter and upon his death turning it off. Romi would change state upon seeing Mercutio's emitter going high when he finishes circling, then when Mercutio dies and turns it off change state again and start hunting Tybalt.

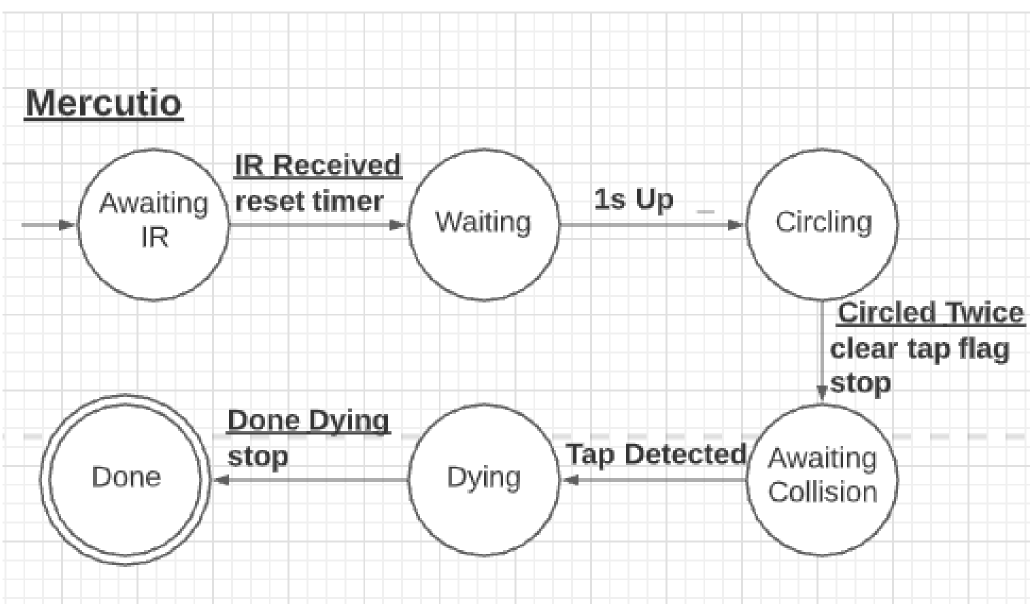


*Figure 4: Fight Scene State Diagram – Mercutio*

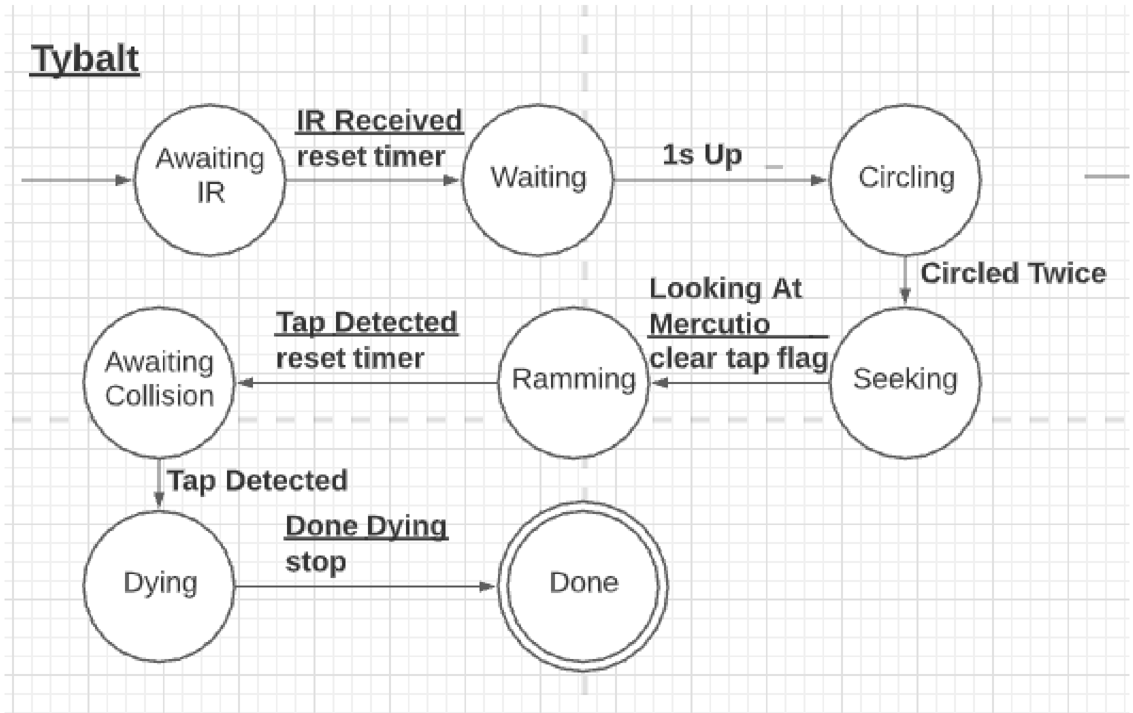


*Figure 5: Fight Scene State Diagram – Tybalt*

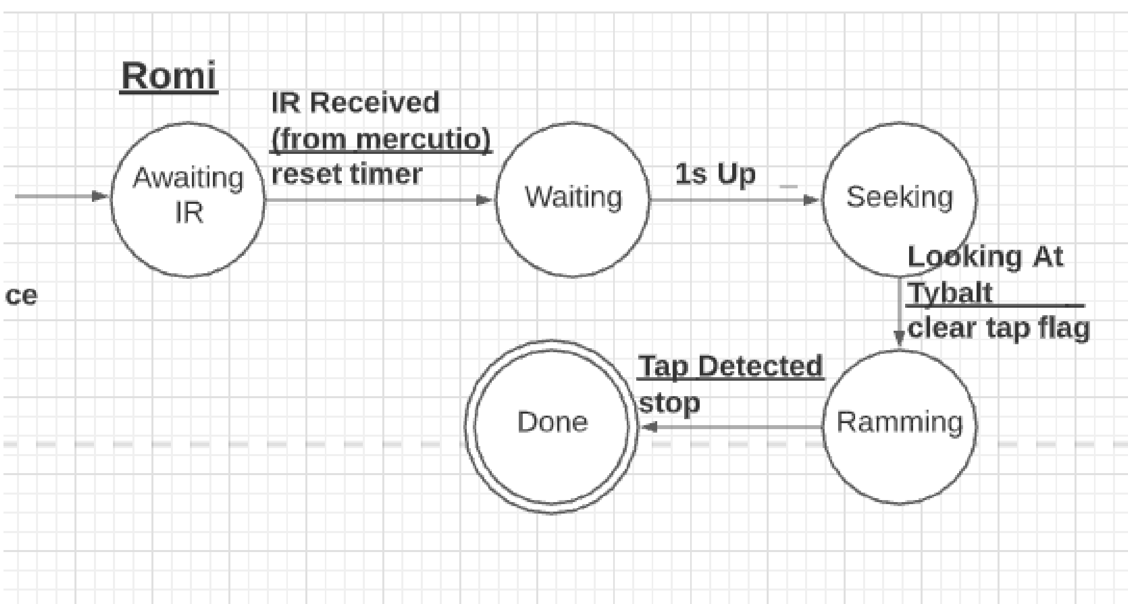


*Figure 6: Fight Scene State Diagram – Romi*

## Balcony Scene

The two greatest challenges involved in recreating the fight scene were getting Juliet to detect the beginning and end of the slope of the ramp and having Romi follow alongside Juliet. Other functionality that had to be implemented includes triggering the robots to start with the IR remote, implementing IR communication between the robots, getting Romi to stop the correct distance away from the ramp and turn, and getting Juliet to stop at the end of the scene with line detection.

For the slope detection challenge, our team took inspiration from Lab 4 to implement a similar system. The inertial measurement unit (IMU) was used to detect the beginning and end of the slope. To get an accurate estimation of the slope, a complementary filter was used to implement sensor fusion, taking readings from both the accelerometer and gyroscope. The complimentary filter is an excellent way to improve the performance of the sensors because it combines the strength of each sensor to mitigate the faults of the other. The gyroscope is most accurate in measuring changes in the pitch angle of the robot, but is susceptible to drift. The accelerometer however is not as accurate in measuring the angular position of the robot, but gives steady readings over time. The complimentary filter works by essentially taking a weighted average of the predicted slope from the gyroscope and the measured slope from the accelerometer. Given that the robot was not expected to experience any large accelerations or large rotation rates, the full-scale range of each sensor was set to the smallest option in order to increase the resolution of the readings. This is a full-scale range of +/- 125 degrees per second for the gyroscope and +/- 2g for the accelerometer. A k value of 0.75 was chosen for the complimentary filter, placing more emphasis on the gyroscope reading, because this yielded good results during testing. To detect when the robot is on the slope, hysteresis was implemented to ensure that the state would not change when not desired. Two distinct thresholds were set to accomplish this. At the start, the robot must register a pitch angle of at least 10 degrees to transition it into the ramp state. The robot then only registers it has left the ramp when pitch angle drops below 2 degrees. This slope detection system with the IMU and complimentary filter was tested in isolation to ensure it would work reliably in the balcony scene.

The other big technical challenge in this scene was getting Romi to follow alongside Juliet as the two robots exit the scene together. A simple method would be to simply set the robots to start at the same time and travel at the same speed, but this was deemed to be an unreliable solution. Alternative ideas included using the IR flame sensor or openMV camera to track the motion of one robot with the other. While the IR flame sensor would have been a good choice to calculate the direction of where the other robot is located, it cannot easily calculate distance from the other robot. The idea of using the IR flame detector along with a second sensor to control the distance between the robots was considered. The team decided instead that the best option would be to use the OpenMV camera to track an April Tag mounted to the other robot. This solution allows for one sensor to give a good estimate of both the direction and distance to the other robot. To accomplish this, the OpenMV camera was mounted to Romi facing to the right side, and an April Tag was mounted to the left side of Juliet. The tag chosen was ID#0 of the 36h11 family. Juliet was programmed to drive straight at a constant speed, while two levels of PID control were used to control Romi’s movement. First, to control Romi’s speed, the x-position of the April Tag was measured relative to the center of the camera’s field of view to determine if Romi was ahead or behind Juliet. A PID controller was used to apply a correction in addition to Romi's base speed in order to keep the April Tag centered in the camera's frame such that Romi would follow alongside Juliet. In addition to this speed controller, a distance controller was implemented to maintain a constant distance between the two robots. This was done by first measuring the width of the April Tag in camera pixels when Romi first turns to face Juliet with the camera. The width of the tag is then continuously measured to determine if the tag is closer or farther away from the set distance. A correction is then added to the speed of one wheel and subtracted from the speed of the other wheel such that the robot will turn accordingly to maintain the set offset distance. This method was tested extensively in isolation by getting one robot to track a handheld April Tag. Many iterations of testing were performed to tune the PID gains until a good controller was produced which would be reliable enough to implement in the final system.

Another feature of this scene is IR communication from the IR remote to the robots, as well as between the two robots. Both robots were equipped with IR receivers to pick up signals, and Juliet was also given an IR emitter LED. At the beginning of the scene, both robots begin in an *AwaitingIR* state in which the IR receiver is continually polled until a signal is detected, triggering a change to the next state. Due to the simplicity of this method and the ability to transmit signals wirelessly over a distance, it was chosen to be the preferred method of communication between the two robots as well. This was implemented in much the same way, with one robot polling the receiver in a waiting state until the other robot signals. In the balcony scene, Romi begins looking for an IR signal at the very beginning of the scene as he waits to approach the ramp. To signal to Romi when it is time to approach, Juliet is equipped with an emitter facing in Romi's direction which is only turned on once she reaches the top of the ramp. The signal is then picked up by Romi, which triggers a state change to begin the approach to the ramp. IR communication is also used at the end of the scene in the same manner. When Juliet stops at the end of the scene, the IR emitter is turned on again to signal for Romi to stop.

During Romi's approach to the ramp, a method had to be determined for stopping the correct distance away from the base of the ramp and turning 90 degrees. Both the ultrasonic rangefinder and Sharp IR sensor were considered as options for measuring the distance from the base of the ramp. The Sharp IR proximity sensor was chosen over the ultrasonic rangefinder because the Sharp IR gives better resolution, and the distances being measured were not expected to exceed the 80cm range of the sensor. The IR sensor was carefully calibrated as was done in the lab activity in order to give the most accurate distance measurements possible. This was done by taking a sample of readings at each of several known distances, and plotting the results to develop a transfer function for the sensor. This transfer function took the form of a power function with constants A = 29.55 and B = -3.56. This transfer function was used to convert the raw sensor input to a distance in cm. For approach state, the robot was programmed to drive forward until recording a distance closer than 15cm, at which point it transitions into a turning state.

For the turning state, the motor encoders on the Romi were used to turn precisely 90 degrees. This was done by writing a method in the chassis class which sets the robot's spin in place and calculates the target number of encoder counts required to reach the desired angle. First, the distance that one wheel must travel is calculated as the circumference of a circle with the robot's wheel track as its diameter multiplied by the fraction of a circle which corresponds to the given turn angle. The number of encoder counts per centimeter driven was also calculated from the wheel diameter and the number of encoder counts per rotation of the wheel. The target encoder count was then set by simply multiplying the distance to be driven in cm by the number of encoder counts per cm. This method is called to tell the robot to begin turning 90 degrees. A second method was written to compare the current encoder counts to the set target counts to determine when the robot has reached the correct angular position. This method is called continuously in the turning state to detect when the turn is complete and stop the robot at the desired heading.

The last challenge for this seen was determining how to make Juliet stop at the desired endpoint. Some ideas to implement this functionality included measuring the distance away from a wall or April Tag, using a light sensor to measure a line or change in surface, or using inverse kinematics to drive a set distance from the bottom of the ramp. Ultimately, the chosen solution was to use the reflectance array sensor to detect a small line made with tape at the desired end location. This was chosen as it was straightforward to implement and allowed our team to demonstrate the use of another sensor. To tune this system, the reflectance sensor was mounted to the front of a robot, and a sample of sensor values were recorded when the robot was positioned on the desk surface and for when the robot was position over the tape. A threshold value was then chosen in between these values, such that the program could compare the sensor value to the threshold value to determine if the robot is on the line. To implement this in the final program, a function was written to return a Boolean indicated whether or not the robot is on a line. As Juliet drives away at the end of the scene, this function is called continuously until a value of true is returned, triggering a state change to the stopped state and stopping the robot. As discussed above, Juliet also uses the IR emitter to signal for Romi to stop upon reaching the line. The reflectance threshold value was calibrated, and the line detection functionality was tested independently before being integrated into the scene to ensure reliability.

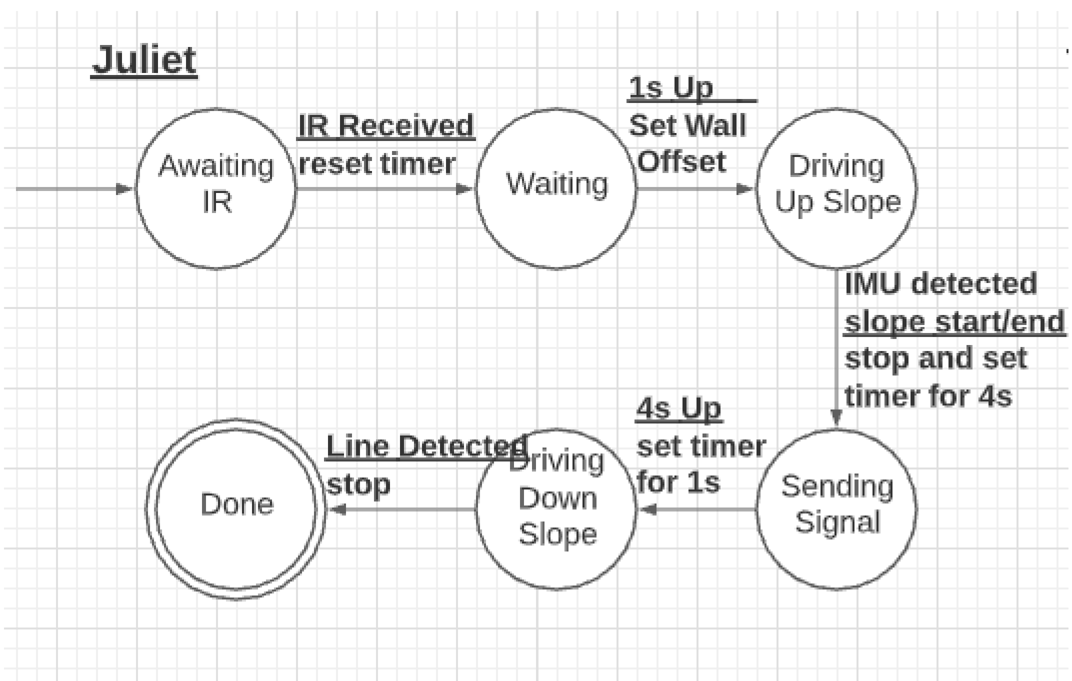


*Figure 7: Balcony Scene State Diagram - Juliet*

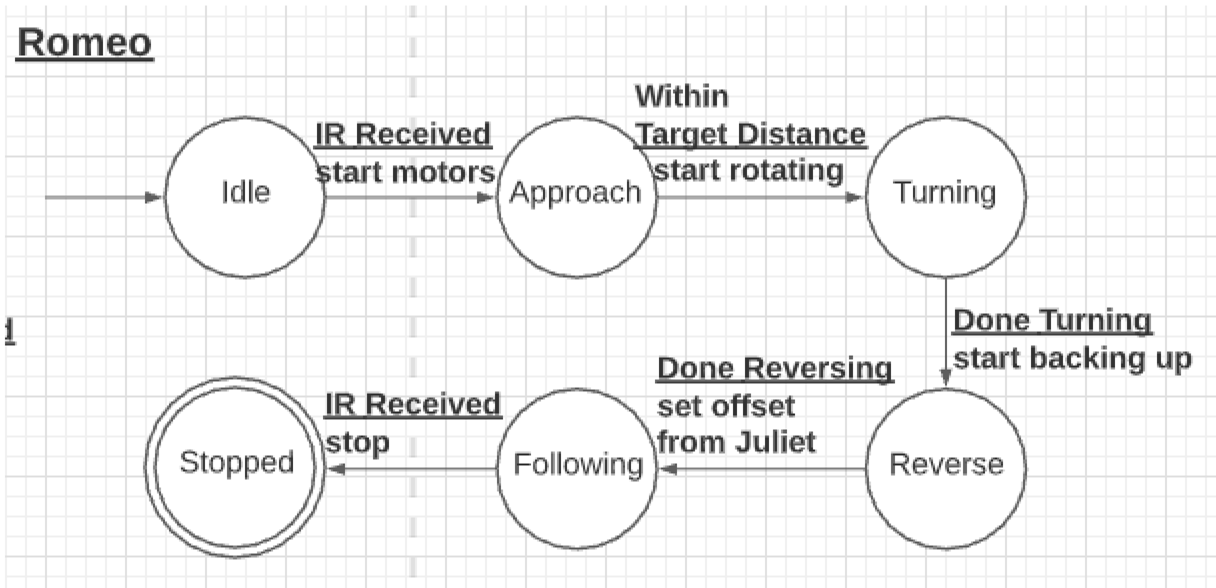


*Figure 8: Balcony Scene State Diagram - Romi*

## Final Scene

Our first challenge for the final scene was for Romi to approach Juliet, and to know when to take his poison. To accomplish this, we used two sensors: the Ultrasonic Rangefinder and the Sharp IR Sensor. Romi utilized the Ultrasonic Rangefinder to perform wall following in order to keep himself on a straight track towards Juliet. While this is happening, Romi utilizes his Sharp IR Sensor to detect the distance he is at from Juliet. Juliet is outfitted with a slip of paper to create a surface in front of her, to make it easier for Romi's Sharp IR Sensor to detect her. Once Romi has reached his target destination from Juliet, he switches states from *Approaching* to *Drinking*. The challenge with "drinking the poison" wasn't how we would accomplish this in software, but how we would accomplish this in hardware. For this part of the scene, we used the micro servo that had a 3D printed attachment to make the servo tip the cup, and a 3D printed cup with a platform base. Here, Romi stops briefly to take a sip from his poison cup. His servo motor activates and tips the 3D printed platform and small cup filled with "poison pellets".

The biggest challenge in the final scene was having Romi be able to kiss Juliet. Unlike the stabbings in the first scene, we decided that the kiss should be soft and gentle. However, because the kiss is soft, we would be unable to use the IMU to detect this contact between the two robots. Unfortunately, we could not figure out how to detect this very slight contact between the two robots using the sensors that we had available to us. Instead, we turned to inverse kinematics in order for Romi to gently give Juliet her final kiss goodbye. Here, Romi drives a set distance of 10 cm forward slowly, stops to deliver the kiss, and backs away. This is separated into three different states: *Dying*, *Colliding*, and *BackingUp*. The *Dying* state includes Romi driving forward to deliver the kiss. Once Romi has reached the 10 cm, he enters the *Colliding* state where he kisses Juliet for 2 seconds. Finally, is the *BackingUp* state where Romi backs away from Juliet slowly, preparing to die. Lastly, the death sequence occurs when Romi finishes backing up. This consists of Romi spinning in a circle coming to a stop, and then lighting his red LED to signal that he has passed away.

The final challenge in the final scene was figuring out how Juliet can tell if Romi has died. To accomplish this, we used the IR communicator on Romi, and the IR receiver on Juliet. When Romi enters his death sequence, right before his red LED is turned on, Romi sends out an IR signal to Juliet to indicate that he has indeed died. Juliet then receives this signal from Romi and goes from her *Waiting* state to her *WakingUp* state. Here Juliet's yellow LED lights up to indicate she has awoken, and she makes a slight turn towards the audience to further indicate her wakefulness. After these actions finish, Juliet enters the final state *Done*, where Juliet remains still and the green LED on the Romi 32U4 is turned on to indicate that the program has successfully and fully run through.

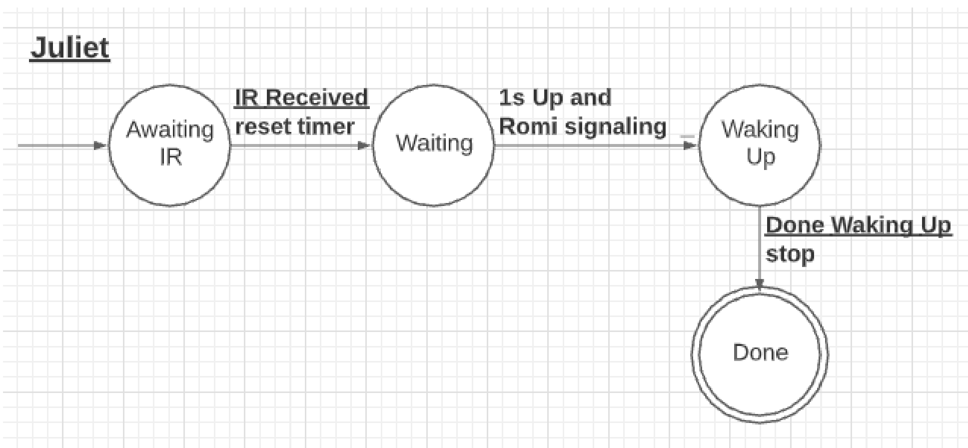


*Figure 9: Final Scene State Diagram – Juliet*

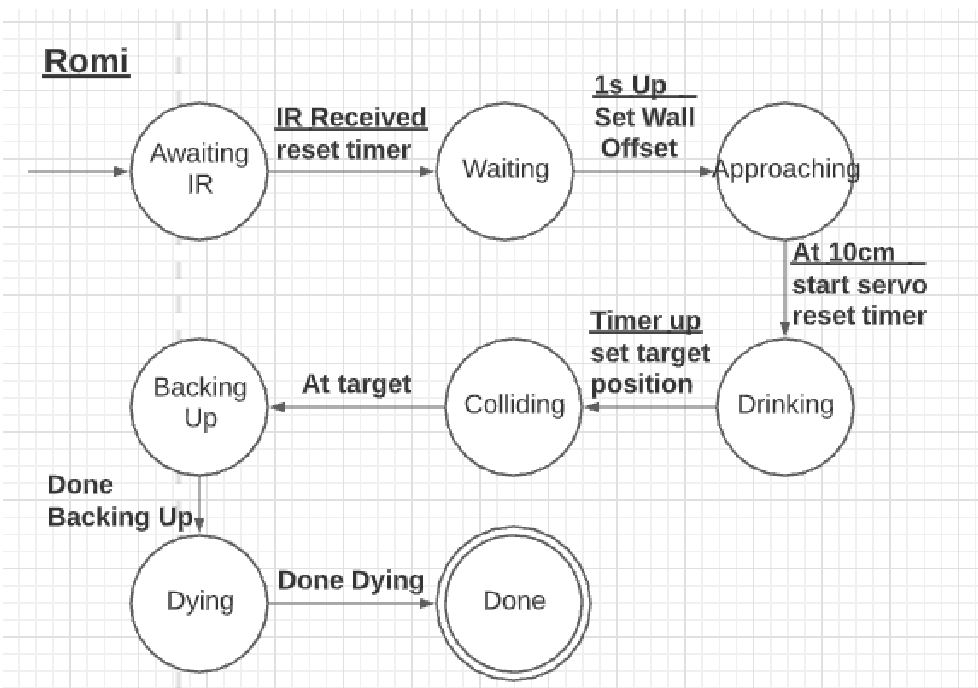


*Figure 10: Final Scene State Diagram - Romi*

# Performance

## Fight Scene

In the demo during the fight scene all three robots were successful in completing their designated tasks. Tybalt and Mercutio both successfully completed two circles before coming to a stop. Both Tybalt and Romi were successfully able to find their target during their Seeking states using the IR and OpenMV cameras respectively. They were both subsequently successful at ramming while maintaining their heading towards their target. All three robots were successful in detecting when they had collided with or had been collided with by another robot. The only error observed in the robots preforming their tasks was Romi entering his seeking state, where he turns until he is looking at Tybalt, early starting shortly after Mercutio started his signal rather than after he was dead and disabled the IR emitter. This is presumably due to the receiver not registering the constant 38khz transmission as always on losing an input during it, a phenomenon observed in lab 4 that was notably bad when there was poor signal quality, due to dimming or distance. This did not affect the final result of the scene. However, it could be mended by altering the criteria for entering the Seeking state from a single low reading to several spread over a certain amount of time, in effect implementing a digital low pass filter for the IR receiver input.

## Balcony Scene

The balcony scene performance was successful in meeting our expectations. Juliet entered the scene and drove up the ramp, successfully detected the flat slope at the top as expected and came to a full stop. The IR communication between the robots was proved successful as Romi was able to pick up the signal sent by Juliet at the top of the ramp and began to approach. Using the Sharp IR sensor, Romi stopped 15cm way from the ramp as specified and turned 90 degrees for the camera to face Juliet. Romi was not perfectly aligned with Juliet upon completing the turn. But quickly picked up the April Tag and adjusted his position relative to Juliet. As Juliet began to descend the ramp, Romi successfully followed alongside her with a controlled movement. Romi was angled slightly away from Juliet initially and drove a bit away, but successfully arced back in towards Juliet to achieve the targe offset distance. Juliet came to a full stop as expected after reaching the strip of tape at the end of her path, and Romi received Juliet's signal to stop alongside her.

Overall, the balcony scene was very successful and met our goals as defined in the design review presentation. The performance was able to demonstrate that all of the functionality described worked as intended. Notably, the April Tag tracking with the OpenMV camera produced a good controller to keep the robots in alignment. The effectiveness of this method was demonstrated in how well Romi both matched Juliet's speed throughout the motion and was able to correct for the distance offset. This aspect of the performance was particularly rewarding after spending a lot of time tuning the PID gains over many iterations to produce a desirable control response. The implementation of the complimentary filter for measuring the pitch angle of the robot also proved effective for Juliet to detect the top of the slope and stop precisely in the small space afforded. The remaining functionality including IR communication, Sharp IR distance detection, turning for a specified angle with the motor encoders, and stopping on a line all proved successful as well as the robots completed the scene as intended. One possible area for improvement might be to implement wall following or another method of keeping straight for Juliet when driving up and down the ramp. This was deemed unnecessary because the robot had no trouble staying straight over the short distance, but this addition would make the system more robust, particularly if there were more variations in the set design.

## Final Scene

During the demonstration, the scene executed successfully and without any flaws. Romi and Juliet both successfully started upon the IR remote control. Romi began wall following until he met his required distance from Juliet, while she patiently awaited Romi while asleep. Once Romi reached his target distance, he activated his servo motor and drank his poison. Slowly he made his way over to Juliet, gave her a gentle kiss, and then backed away. He then entered his death sequence where Romi completed one full rotation, backed away from Juliet, spun, and finally stopped and activated his red LED to signal that he died. Immediately after, Juliet then woke up, shined her yellow LED to signal her wakefulness, and turned slightly towards the audience.

During this final scene, the robots both completed their tasks flawlessly. Romi was able to successfully wall follow to keep himself straight on his path to Juliet. He did not appear to be at a slant, and he did not turn at all during his approach. Next, the Sharp IR Sensor was able to successfully detect Juliet at her correct distance. Romi was able to stop at the appropriate distance from her to drink his poison, instead of stopping either too far behind or too close to her. Next, the inverse kinematic for the kiss were perfect for the final scene. In tests, Romi has shown to stop too far away from Juliet, so that they do not successfully kiss. Romi has also shown to not stop at Juliet too, where he can push her for a distance until he enters the *Colliding* state. However, in our final scene Romi was able to drive the correct distance to Juliet, where he made light and gentle contact with her before dying. Finally, Juliet was able to receive the signal from Romi that he was dead flawlessly. In past tests we have experienced times where Romi's IR Communicator was wired wrong, or when Juliet's IR Receiver was blocked and could not receive the signal. However, in the demonstration, Juliet was able to quickly receive Romi's signal of death and immediately enter her *WakeUp* state. Additionally, both robot's LED lights worked perfectly for the demo as well. Each light remained off until their respective cues to turn on. Specifically, Romi's red LED only lit up until after he had completed one full revolution in his death sequence and came to a stop, to indicate that he had died. For Juliet, her yellow LED was not lit until she had received the signal from Romi to indicate that he had died. As soon as she received the signal, her yellow LED lit to indicate that she had awoken, and then she proceeded to complete her slight turn towards the audience.