Aerojet Rocketdyne Remote Controlled Rover Payload

Chris Keefer, Joel Marquez, Anjali Padmanaban, and Chase Walker

Dept. of Electrical Engineering and Computer Science, University of Central Florida, Orlando, Florida, 32816-2450

Abstract **— This paper presents the design of a payload created for the purpose of deployment from a rocket for a competition. The payload is a combination of a capsule and rover with remote control and wireless video capabilities. The payload has autonomous deployment functionality, and is designed for long-distance use. The payload and capsule designs were motivated by the rules and requirements of the FAR 51025 competition.**

I. Introduction

This project was designed around and motivated by the rules of the Friends of Amateur Rocketry (FAR) 51025 competition. The competition goal is to make a rocket and accompanying payload to score the most points as outlined in the rules. Aerojet Rocketdyne provided the University of Central Florida with funding to send one full team - rocket and payload - to compete in this competition. This project is one of the multiple senior design payload projects competing to have the competition spot. This team developed the payload while working with the team that is responsible for the rocket.

The FAR competition outlines multiple payload options. The payload chosen for this project is intended to serve as a reconnaissance vehicle for its landing location. This payload is required to be remote controlled, output live video to the control station, and travel a minimum of 10 feet from its landing location in the desert.

The deliverables of this project are:

1) A capsule that provides a container for the rover during launch, and protection for the rover during landing.
2) A rover capable of remote controlled operation and live video transmission.
3) A control station capable of operating the rover and receiving its video transmission.

Each of these deliverables has a circuit board responsible for its function. The capsule PCB is responsible for control of the rover-capsule coupler. The rover PCB powers the video transmission and remote control reception components, and is responsible for rover motor power and control. The control station PCB will power the video reception and remote control transmission components, and provide the logic for transmitting the joystick input to the rover through the transmission hardware.

II. System Overview

The three deliverables of this project exist as the three main subsystems required for payload function. These subsystems are the capsule, the rover, and the rover control station.

A. Capsule

The capsule is a core component to the payload and is used to house and protect the rover from the environments it is expected to experience throughout flight and landing. The capsule interfaces with the parachute and the rocket payload bay using eye hooks and railings respectively. It also features an electromagnetic lock which will be used to couple the rover to the inside of the capsule.

The capsule PCB contains sensors that will be used for landing detection to allow the rover coupling to be disengaged. This is part of the Capsule Landing Automated Sequence System (CLASS) and will be explained in depth later.

B. Rover

The rover is a land vehicle designed to fit into the capsule, and to traverse desert terrain. The rover is capable of forward and backward movement as well as pivot turning via skid steer style turning. The rover uses 4 DC brushless gear motors as its powertrain. Custom-made wheels are used with these motors to allow the rover to traverse concrete-like land and sandy land.

For video transmission, the rover uses a 5.8GHz system to send real-time video to the rover control station. For rover control reception the rover uses a 2.4GHz receiver to capture the rover control station commands.

The rover also contains the same sensors as the CLASS system in the capsule to provide the same autonomous operation via landing detection.

C. Rover Control Station (RCS)

The rover control station serves as the base for rover communication. The RCS contains the joystick, radio transmitter, video receiver and display screen. The analog joystick is used to control the rover from the control station. The radio transmitter uses 2.4GHz frequency to transmit commands to the rover. These commands depend on the joystick's coordinates. The video display will be used to display the live video stream that the rover transmits during its mission. This video stream is received by a 5.8GHz video receiver.

III. System Concepts

The payload has multiple interacting systems with their own complexities. The concept of each subsystem can be presented with details of design and of subsystem interaction to provide the overall view of the payload's function. Fig. 1 gives an overview of the entire system in block diagram format.

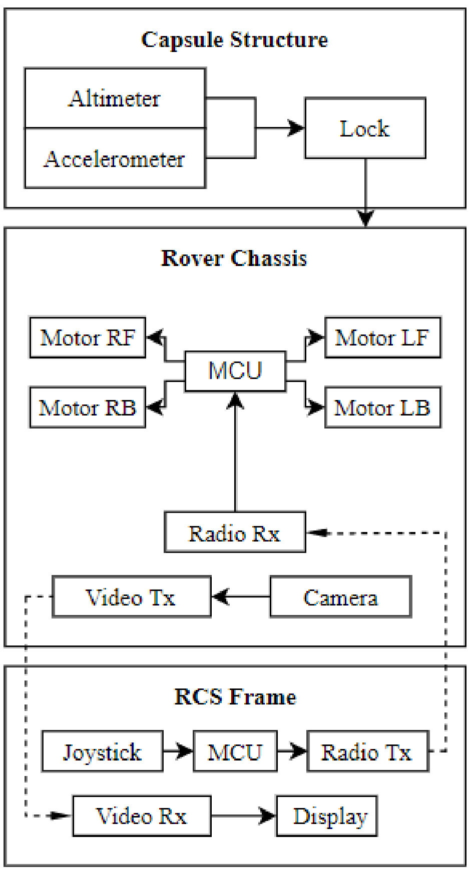

Fig. 1. Full system block diagram that shows inter and intra-system interactions.

A. Capsule

1) *Structural Design:* The main working material for the capsule was selected to be AISI 1060 carbon steel for its superior mechanical strength, low cost, and workability. This steel can be easily welded, cut, and drilled through with ease. The structure of the capsule is that of a cylinder-like container with a hexagonal cross section. The sides of the capsule were welded together using an oxygen-argon torch welder. The top of the capsule consists of bent sheet steel that is riveted to the rest of the capsule structure. The front of the capsule features a scooped geometry where the top and sides of the capsule meet at a point above and forwards of the bottom of the capsule. This scoop feature is a key feature to the reliable landing system developed for this payload.
2) *CLASS:* This is a key system that allows the rover to leave the capsule when it lands. The CLASS is responsible for releasing the coupler that keeps the rover secure during the landing procedure. The CLASS system uses an accelerometer to detect when the capsule has landed after liftoff. The logic of this system is shown in Fig. 2. When landing is detected, the electromagnetic coupler will be released using a load switch and the rover will be free to leave the capsule.

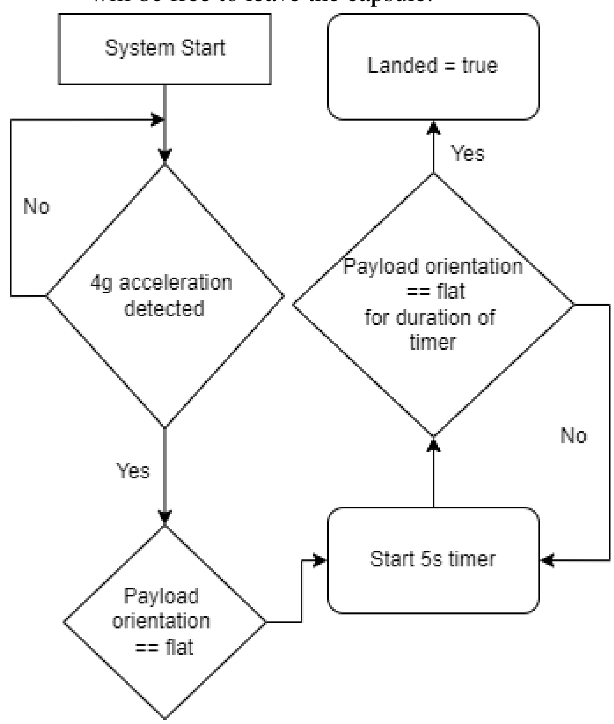

Fig. 2. Flowchart that illustrates the logic of the landing detection system from pre-launch power on to landing.

B. Rover

1) *Chassis Design:* The material composition of the rover chassis is a very important design consideration as the rover must be able to withstand shock and vibrations of varying frequencies throughout its mission. The material must be strong and rigid yet flexible enough to

successfully transmit vibrations without causing damage to the rover chassis itself or the electronics stored on board. The material selected for the chassis must be a thin metal that is weldable and easy to drill through. The chassis will hold the motors and all of the electronics that allow the rover to operate, and it therefore must also accommodate and provide protections for these components. AISI 1060 carbon steel was selected for its mechanical strength, weldability, and ductility. This steel was selected for the purposes of its workability as well as cost.

2) *Rover Propulsion:* The decision was made to utilize differential steering in the design of the rover for this mission. The chief advantage to this style of steering is its simplicity. This steering option uses few moving parts which reduces the mechanical complexity of steering the rover. Having fewer parts to the steering system also works to reduce the overall weight of the payload. A secondary advantage of this configuration is ease of control for the user in command of the rover. Simple power inputs can be made to control the rover and spin it in place to record its surroundings. As for the wheel setup, the decision was made to go with the continuous track configuration which is compatible with the differential steering system. This decision was made due to the overwhelming performance advantages held by continuous track vehicles in sandy or desertlike environments. The rover also does not need to be able to achieve high speeds or turn at speed, therefore this design will move forward with a continuous track wheel system.
3) *Rover Auto-Start:* Much like the capsule has the CLASS, the rover has an auto-start system that allows the rover to autonomously leave the capsule after landing. The rover also uses an accelerometer to detect landing in the same manner as the CLASS, see Fig. 2. However, when the rover detects landing, it will wait for 2 seconds for the coupler to be disengaged, and then drive itself out of the capsule. The rover must do this autonomously because the remote control signal will not be able to penetrate the steel capsule. Once the rover has exited, it can be controlled by the user.
4) *Remote Signal Reception:* The radio receiver located on the rover will receive commands from the radio transmitter on the RCS. Universal Asynchronous Receiver/Transmitter (UART) is used to allow serial communication between the receiver and transmitter.
5) *Rover RC Movement Design:* When the RC signal is received, the rover needs to decode the signal into one of four movements: forward, backward, left spin, and right spin. The motors chosen have both speed and direction controls that will be used to generate these four movements. The motor speed and direction are controlled via MCU using PWM for speed and GPIO for direction control. As mentioned before, skid steer will be used and is achieved via turning the pairs of motors on either side in opposite directions. It should be noted a decision was made to not give the user granular speed control of the motors and therefore only a zero or one hundred percent duty cycle for the PWM speed control will be used. This decision was made because the motors do not move at a high enough speed to be uncontrollable in any traversal situation.
6) *Video Transmission:* The video transmitter for the rover operates in the 5.8GHz frequency band. This frequency was chosen because it does not interfere with frequencies being used by the avionics. This transmitter will also allow for a fast video transmission at the estimated distance of 1 km. Another deciding factor in the frequency band selection was the lack of 2.4GHz video transmitters and receivers due to the global chip shortage. The antenna chosen is the omnidirectional Pagoda antenna. This antenna allows for omnidirectional signal transmission, this will help with signal transmission and reception because the 5.8GHz band is prone to dropping signal suddenly.

C. RCS

1) *Video Reception and Display:* The goal of the video reception is to provide a live view for the rover operator. The only constraint for the video signal receiver was that it be compatible with the video transmitter and receive the signal over the approximately 1km distance the rocket will have traveled from the start point. To reach the required distance with the video receiver, it was found that a pagoda antenna was required for omnidirectional reception. The display of the live feed only needs to be good enough for the operator to avoid obstacles. The target specifications for the display were a resolution of 480p and a frame rate of 24 frames per second.

The rover currently operates at approximately 480p resolution and 30 frames per second.

2) *Remote Control Transmission:* The remote control transmission is used to send the rover commands on which direction to move. The remote control uses an analog joystick to control the rover. The coordinates of the analog joystick's position are categorized into four commands: forward, backward, left and right. The radio transmitter uses Attention (AT) commands to communicate with the radio receiver. These commands are then transmitted to the radio receiver located on the rover through UART serial communication.

IV. Hardware Detail

The concepts previously described, drive the function of the payload, but the hardware behind these concepts is crucial to the function. The impactful hardware needed for replication of the payload design will be listed below with details of use and hardware specifications.

A. Motors and Wheels

The wheels were created out of ABS plastic drive gears. These gears were selected to make the wheels because of their form factor and cheap cost. The teeth of these gears are ideal for gripping into the ground and gaining traction in rugged terrain.

The wheels have a diameter of 40 mm and a circumference of 125.6 mm which allows the speed of the rover to be calculated. At a rotational speed of 80 rpm, the rover will move at .16 meters per second (roughly .5 feet per second) as seen in (1).

$$\frac{125.6\,mm * 80\,rpm}{1000\,mm/m * 60\,sec/min} = .158\,m/s \qquad (1)$$

B. Electromagnetic Coupler

The electromagnetic coupler is the component that keeps the rover secured inside of the capsule during descent. This coupler consists of a metal housing with a cable for power, and a metal stem that locks into the housing. The lock is rated for 50kg of holding strength which is strong enough to survive parachute deployment.

Parachute deployment is expected to induce 8g of force on the capsule. The fully assembled rover weighs 800g which means the lock will only experience 6.4kg of force as seen in (2).

$$800grams\ *\ 8g\ =\ 6.4kg \qquad (2)$$

The housing of the coupler is bolted to the back wall of the capsule where the parachute attaches. The stem is bolted to the back of the rover. During descent this causes the rover to be suspended with the front facing the ground, and the front of the rover closest to the exit.

The coupler is normally closed, meaning power is only needed to open it. This was chosen for both power savings and if power fails, the rover will be safe even if it will not be able to leave the capsule. The coupler unlocks with 12V and 100mA of power.

C. Video System

The video transmitter and receiver chosen for this project are the Wolfwhoop WT832 and WR832. This transmitter/receiver combination can deliver reliable video signal across 40 different 5.8GHz frequencies. The transmitter module takes in 12V from a power supply and has 3 wires to connect the camera to. There are two wires for 12V power to the camera and one for video data. After receiving the video data from the camera, the transmitter sends the signal through the antenna using Wideband Frequency Modulation. Frequency Modulation helps with signal stability compared to its Amplitude Modulation counterpart. The transmitter and receiver both come with Whip antennas, but whip antennas can lose signal easily if they do not have the same orientation. Instead the antennas chosen for the transmitter/receiver combination are the Pagoda omnidirectional antennas. These antennas allow for a circular polarization of the signal, which means the rover operator does not need to focus on the angle between the video transmitter and receiver. The receiver is also powered with 12V and has an output for an AV connection. This AV wire gets plugged into the monitor straight from the receiver.

The display chosen is the 5 inch GreenYi-08 LCD Color screen. The display has a 800x480 resolution which will provide the rover operator with enough visual context to make informed decisions about navigation. The input for the monitor is also AV which makes the display directly compatible with the output of the receiver. The input voltage required for the monitor is also 12V allowing for less voltage conversions.

D. Remote Control System

The radio transceivers chosen for this project is the REYAX RYLR896 Lora Module. This module features the Lora long range modem which provides ultra-long range spread spectrum communication and high interference immunity while minimizing current consumption. By default, the module uses a radio frequency of 915MHz and bandwidth of 125KHz. The module is easily controlled using AT commands. AT commands are instructions commonly used to control modems. Types of AT commands include set, read, and execution. Set commands can be used to change various parameters like spreading factor, coding rate, or network ID. Read commands are used to see what parameter values are already set, and execution commands can be used to send messages. The syntax for sending a message through this module is AT+SEND=<Address>,<Payload Length>,<Data> followed by a carriage return and new line. A received message reads as +RCV=<Address>,<Length>,<Data>,<RSSI>,<SNR>.

E. MCU

The Texas Instruments MSP430FR6928 is the MCU used in all three of the circuits that power the subsystems of this project. This MCU is a 16-bit RISC MCU with an up to 16-MHz clock.

This MCU was chosen for multiple reasons:

1) The University of Central Florida's Embedded Systems class gave the two CpE members and one EE member large amounts of embedded systems practice with this MCU. This choice allows the members to save time learning new coding schema for another processor. Another benefit of this MCU choice is the members have access to the Embedded Systems Lab Manual provided by the Department of Electrical and Computer Engineering. This lab manual is a detailed resource that was used as a refresher on MSP430 programming.

2) The three ECE team members have MSP430FR6989 Launchpads that allow the MCU circuits to be programmed through Texas Instruments Code Composer Studio. A launchpad acts as a bootloader for the MCUs, allowing them to be used instead of extra hardware being purchased.

3) Texas Instruments provides very detailed documentation for both using the chosen MCU on a PCB, and programming the MCU. Knowing the quality of this documentation allowed the team to choose this MCU with confidence that time would not be wasted reading poor documentation to use it in the system designs.

F. Load Switch

A load switch is an IC that acts as a physical throw switch, but is a series of MOSFETS that can be controlled with one MCU signal. A Vishay Electronics Si1869DH was chosen. It was the best choice because it supports the 12V throughput it will experience with low on resistance.

The CLASS PCB uses the load switch to control the electromagnetic coupler. Since the coupler is normally closed (NC), it does not need power to stay locked. Once landing has been detected by the accelerometer, the load switch will be "closed" via a high-level output from the MCU to direct 12V to the lock, freeing the rover.

G. Accelerometer

The accelerometer chosen is an Analog Devices ADXL343 3-axis capacitive accelerometer. It provides range settings of +/-2, 4, 8, and 16 G with 10-bit precision. This sensor will be used via I2C.

V. Software Detail

The software behind this project is critical to its success. The main systems designed are landing detection, coupler release, rover auto-start, and remote control. To understand the software for this project, the whole expected operation must be explained chronologically.

To begin, the payload containing the CLASS PCB and the rover with its PCB, will be loaded into the rocket with both circuits powered on. The coupler will be locked by default, binding the rover to the capsule. The RCS board will also be on if desired, but will not be useful until the capsule has landed.

The CLASS and rover will both be polling for the initial rocket liftoff event to begin their landing detection. Once landing is detected by the CLASS, it will unlock the coupler. The rover will then autonomously turn on all four motors to leave the capsule. It must leave the capsule by itself so that the remote control and video connections can be established.

The RCS will have been powered on pre-launch or during the capsule's parachute assisted descent. Once the rover exits the capsule, the video and radio connections will be initialized automatically. When the RCS receives the video signal, the operator will know the rover can be driven. Rover movement is controlled with the joystick that is in the RCS system. Stationary, forward, backward, left spin, and right spin are the supported movements.

A. Landing Detection

The flowchart overview of this system can be seen in FIg. 2. The details of the system will be described here.

The liftoff event will be confirmed at the time that the accelerometer has seen 4G of X-axis acceleration (the rover will be sitting vertically in the rocket).

For the function of landing detection, it was determined that I2C communication from the sensors would be fast enough to deliver a prompt landing status. With a maximum clock speed of 400KHz the accelerometer, the speed is more than enough to handle gathering readings from the sensors every 5 seconds.

The accelerometer values will be read across I2C communications through pins 1.6 and 1.7. The accelerometer will be configured to its maximum acceleration range of ±16g. The x, y, and z axes each have

2 bytes of data registers where the acceleration value is given in 2s complement format.

When liftoff is detected, the accelerometer will begin checking for the payload to be in a flat orientation (sitting on the ground). A timer will then start to ensure this flat orientation was not due to the rocket's change of direction at apogee. When the timer finishes and does not find the state to be different, landing has been detected. The code implementation of this is seen in Fig. 3.

```
for (;;)
{
    i2c_read_accel_values(ADXL_343, 0x32, &z_raw);

    // If flat
    if (abs(z_raw) > Z_THRESHOLD)
    {
        // wait 1s
        while((TA0CTL & TAIFG) == 0) {}

        // reread the accelerometer values
        i2c_read_accel_values(ADXL_343, 0x32, &z_raw);

        // if still flat after 1s, release lock
        if (abs(z_raw) > Z_THRESHOLD)
            P6OUT |= SOLENOID_LOCK;

        // Clears the flag
        TA0CTL &= ~TAIFG;
    }
    // Else (not flat), lock the capsule lock
    else
        P6OUT &= ~SOLENOID_LOCK;
}
```

Fig. 3. This code shows the MSP430 Timer_A PWM configurations for the motor PWM signals.

B. Coupler Release

The coupler release code is very simple, and only relies on using a singular GPIO pin to "close" the load switch that is "open" and blocks power to the coupler by default.

All that needs to be done is to set any GPIO pin as output, and then send a high signal to the load switch as seen in Fig. 4.

```
// Divert pin 6.0 to GPIO output for load switch
P6DIR |= LOCK_SW;
P6SEL1 &= ~LOCK_SW;
P6SEL0 &= ~LOCK_SW;

// close load switch, unlock coupler
P6OUT |= LOCK_SW;
```

Fig. 4. This code shows the use of an MSP430 GPIO pin to unlock the coupler by closing the load switch it is connected to.

The load switch will be held "closed" for 30 seconds to allow the rover ample time to leave the capsule before "opening" to stop powering the lock.

C. Motor Control

The motors use two GPIO pins and two PWM pins for control of the left and right motor banks direction and speed respectively.

The PWM signals come from Timer_A_0 channel 0 and Timer_A_1 channel 0 for the right and left PWM signals. Both timers are using SMCLK at 16MHz in UP mode. SMCLK was chosen for the signal since the motors require a PWM frequency between 20KHz and 30 KHz. The timers use a reset/set output pattern with TACCR0 set to 837 for a 20KHz signal as seen in Fig. 5.

```
// Starting timer 1 in the up mode; SMCLK
TA1CCR0 = 837; // // @ 16 MHz --> 20KHz
// Starting timer 0 in the up mode; SMCLK
TA0CCR0 = 837; // @ 16 MHz --> 20KHz

// timer 1 (ACLK @ 32 KHz) (Divide by 1) (Up mode)
TA1CTL = TASSEL_2 | ID_0 | MC_1 | TACLR;
// timer 0 (ACLK @ 32 KHz) (Divide by 1) (Up mode)
TA0CTL = TASSEL_2 | ID_0 | MC_1 | TACLR;

// Configuring timer 1 Channel 0 for left motor PWM
TA1CCTL0 |= OUTMOD_3; // Output pattern: Reset/set
// Configuring timer 0 Channel 0 for right motor PWM
TA0CCTL0 |= OUTMOD_3; // Output pattern: Reset/set
```

Fig. 5. This code shows the MSP430 Timer_A PWM configurations for the motor PWM signals.

The GPIO pins for the motors are set as output. Setting a pin high sets a pair of motors to rotate clockwise. Setting a pin low sets a pair of motors to rotate counterclockwise. This can be used to create all four of the rover's supported movements.

D. Rover Auto-Start

When landing has been detected, a flag will be set and the rover must drive itself out of the capsule. To escape the capsule, the rover will wait two seconds after the flag is set to ensure the coupler has been released. It will then drive forward for five seconds to leave the capsule. Five seconds of motor drive results in roughly .8m of movement as seen in (1). With a capsule length of 43cm, the rover is guaranteed to move itself out of the capsule with this amount of movement. After it has done so, the remote control and video links will be established, and the remote control system can be used.

E. Joystick Reading

The two-axis analog joystick works as the combination of two potentiometers that represent the X and Y axis. It works by reading the voltage values through the potentiometer and sending analog values that represent these voltages. To read these values an Analog-to-Digital Converter (ADC) is used. The ADC converts the analog input into a binary number through multiple voltage comparisons.

To read the joystick values, the joystick pins are mapped to the necessary pins on the microcontroller. The X-axis pin is mapped to A1/P1.1 and Y-axis is mapped to A2/P1.2. The default functionality of these pins is GPIO so they are diverted to analog functionality.

To configure the ADC12_B module in order to read the joystick's horizontal and vertical direction, FR6xx Family User's Guide (slau367o) was referenced to initialize the ADC control registers. The ADC12_B module will convert the two analog inputs from the joystick and drop the results into registers ADC12MEMx (x:0,1). Each result register also has its own configuration register ADC12MCTLx (x:0,1) which is configured to the respective analog channels A1 and A2. The X-axis values are then stored in ADC12MEM0 and the Y-axis values are stored in ADC12MEM1. THis can be seen in Fig. 6.

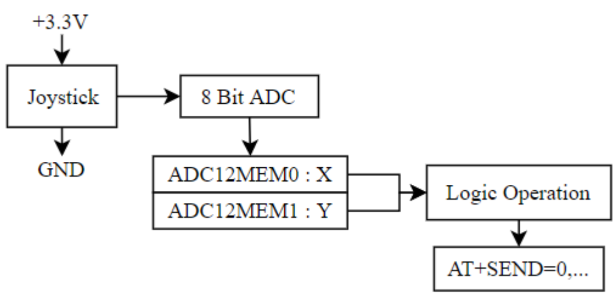

Fig. 6. This code shows the block diagram for remote control transmission starting from the joystick, and to the transmitter.

F. Remote Control

In order to transmit directional commands to the rover, the joystick registers must first be read to determine what direction the rover must move. Integer variables x and y are used to read the values from registers ADC12MEM0 and ADC12MEM1 respectively. The coordinate values that correspond to each direction are shown in Fig. 7. The letters F, B, L, R, and S are used to signify forward, backward, left, right, and stationary. If-statements are used to determine which letter must be transmitted based on the joystick coordinate values.

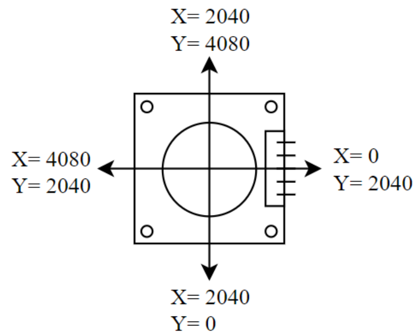

Fig. 7. This is a diagram of the joystick coordinate outputs.

The radio transmitter uses UART to communicate with the receiver through AT commands. The RXD and TXD pins of the transmitter are routed to pins P4.3 and P4.2 which are diverted to UART functionality. The default SMCLK clock is used and configured to use a baud rate of 115,200 with no oversampling. The eUSCI hardware handles the UART transmission and reception of data and interfaces using a few registers and flags shown in Table I below.

TABLE I
REGISTERS AND FLAGS FOR EUSCI UART TRANSMISSION AND RECEPTION

Flag (1-bit)	Register
-	UCA0TXBUF
-	UCA0RXBUF
UCTXIFG	UCA0IFG
UCRXIFG	UCA0IFG
UCRXIE	UCA0IE

When the module is ready to transmit, the transmit flag is 1. To transmit a byte, the byte is copied to the transmit buffer causing the transmit flag to become 0 automatically and begin transmission. When transmission is complete the transmit flag goes back to 1.

To send a command through the radio module the AT command must read as follows: AT+SEND=0,1,F\r\n. This command tells the transmitter to send the character F to any receiver on the same network. To transmit this string successfully the function uart_write_string() is created to send each character byte by byte through UART communication.

The radio receiver receives the message as follows: +RCV=101,1,F,-38,40. To receive this message through UART an interrupt operation is used. Each time a character is received and loaded into the UCA0RXBUF register, the interrupt flag UCRXIFG is raised. The code jumps to the Interrupt Service Routine (ISR) where the received byte is read from UCA0RXBUF and stored into an array to allow the next byte to be received. Once the whole message is reconstructed into the array the interrupt is complete and the normal code will resume.

The received strings will always be the same length, making it easier to identify which letter was transmitted from the RCS. The '+' is stored into the first position r[0] of the array. The direction letter will always be located in r[11]. To determine how the motors must be controlled, r[11] will be read to determine if it is F, B, L, R, or S. These represent forward, backward, left, right, and stationary. Using these values the motor direction or speed will be configured to cause that movement in one of the five following ways:

1) For forward movement the left motors move CW and the right motors move CCW.
2) For backward movement the left motors move CCW and the right motors move CW.
3) For left spin the left motors move CW and the right motors move CW.
4) For right spin the left motors move CCW and the right motors move CCW.
5) For no movement, the pwm duty cycles for all motors will be set to zero percent.

VII. Circuit Board Production Detail

All circuit boards used in this project for the rover, CLASS, and RCS are custom designs made by the members of this team.

The circuit boards were designed using AutoDesk EAGLE. The circuits use a mix of EAGLE libraries as well as UltraLibrarian library downloads. The voltage regulator designs were created by Texas Instruments' WEBENCH software and transposed to EAGLE.

The printed circuit boards were manufactured by JLC PCB in China. All SMD parts were sourced from Mouser, Digikey, or Texas Instruments. Circuit board assembly was done free of charge by Quality Manufacturing Services in Lake Mary, Florida. Their partnership with UCF allowed the team to have free and reliable assembly with very fast turnaround times. All circuit board testing was done using UCF provided equipment.

VII. Conclusion

This project led to the successful creation of a remote control rover payload and landing system. The payload was successfully able to integrate with a rocket and survive the launch on 4/16/2022. All requirements that were set for completion were met by the execution of this project. The members of this project feel greatly accomplished in completing this project and having a fully-functional product in the end.

Acknowledgement

The authors wish to acknowledge the help of Aerojet Rocketdyne, the University of Central Florida, Dr. Lei Wei, Dr. Felix A. Soto Toro, Quality Manufacturing Services, and Kevin Wydra.