ECE 6780 MINI-PROJECT FINAL WRITEUP

Team Name: Olivedrab

Team Members: Nicholas Western, Dmitry Panin, JT Herrmann, Braden Brown

Overview:

In this project we built a rover with two tasks to complete:

1. Obstacle Course- Our rover needed to navigate a course with objects placed as obstacles throughout. We needed to steer the rover through the course without striking the obstacles and without being able to see our rover.
2. Drag Race – Our rover attempted to drive as fast as possible to a finish line and stop as close to the finish line as possible, again, without us being able to watch the rover while it drove.

In addition to the two tasks mentioned above; the obstacle course and the drag race, we also had to design our rover with the following constraints in mind:

1. Our only connection to our rover was a wireless USART or RS232.
2. Our rover had to incorporate at least three topics we had covered in labs out of the following list: GPIO, Interrupts, Timers, UART, I2C, Analog, and PID.
3. Because our group consists of graduate students, we were required to use PID along with three other options.

Design:

A. Components

Our idea was to use an STM32F072 discovery board as our main controller. The topics we implemented from the labs, in addition to PID, was GPIO, I2C, Analog, UART, and Timers. The GPIO was used as the main interface for the sensors we selected. I2C was used to talk with a color sensor used for the drag race aspect of the project; Analog, or the ADC peripheral, was used to talk with the IR sensors (their output is an analog voltage), and UART was used to talk with the ultrasonic sensor's UART output. We incorporated PID with the motors to control the acceleration of the rover both for speeding up and down during the drag race and while avoiding obstacles.

For the body of the rover we used a prebuilt chassis (see BOM below) with two wheels that were hooked up to the motors/motor drivers and a third universal stabilizing wheel was used on the front of the rover. We placed an IR sensor on the right side of the chassis and the ultrasonic sensor on the front of the rover for obstacle detection. Our plan was to hug the right side of the obstacle course and use the right and front sensors to navigate obstacles. We also placed the color sensor on the bottom of the rover for detecting the finish line for the drag race. We powered the rover using a 12V battery pack for the motors and a 6V battery back for the STM32F0 Discovery board, both of which can be found below in the BOM. Also, a photo of our finished rover can be seen below in Fig 1.

B. Design updates and changes

We had to make a few design changes as we went. The first design change was our chassis. We had selected an aluminum chassis at first but the chassis didn't support our motors and we wanted to use motors that we were familiar with and that supported PID control. We also thought we would use interrupts at the start of the project for stopping and navigating the obstacle course but we figured out that our USART3 and USART4 peripheral shared the same interrupt handler and wouldn't allow us to poll on the USART3 and send messages through our Bluetooth over USART4 simultaneously. We also needed the pins that had USART1 and USART2 for other components so we elected to use a purely polling structure. We started the code for autonomous driving but time constraints prevented us from finishing this implementation. Additionally we originally planned on using motor drivers that we designed class but our driver's became inoperable through testing so we had to switch to commercial versions. Our final major design change was with our IR sensors that we used on the sides of the rover. In the end we couldn't get our code to read accurate data from both sensors at the same time. We think this was a sampling issue or an issue of using the one ADC peripheral on our board for two channels. We think the fix would be configuring the ADC peripheral differently or using a board with more than one ADC peripheral.

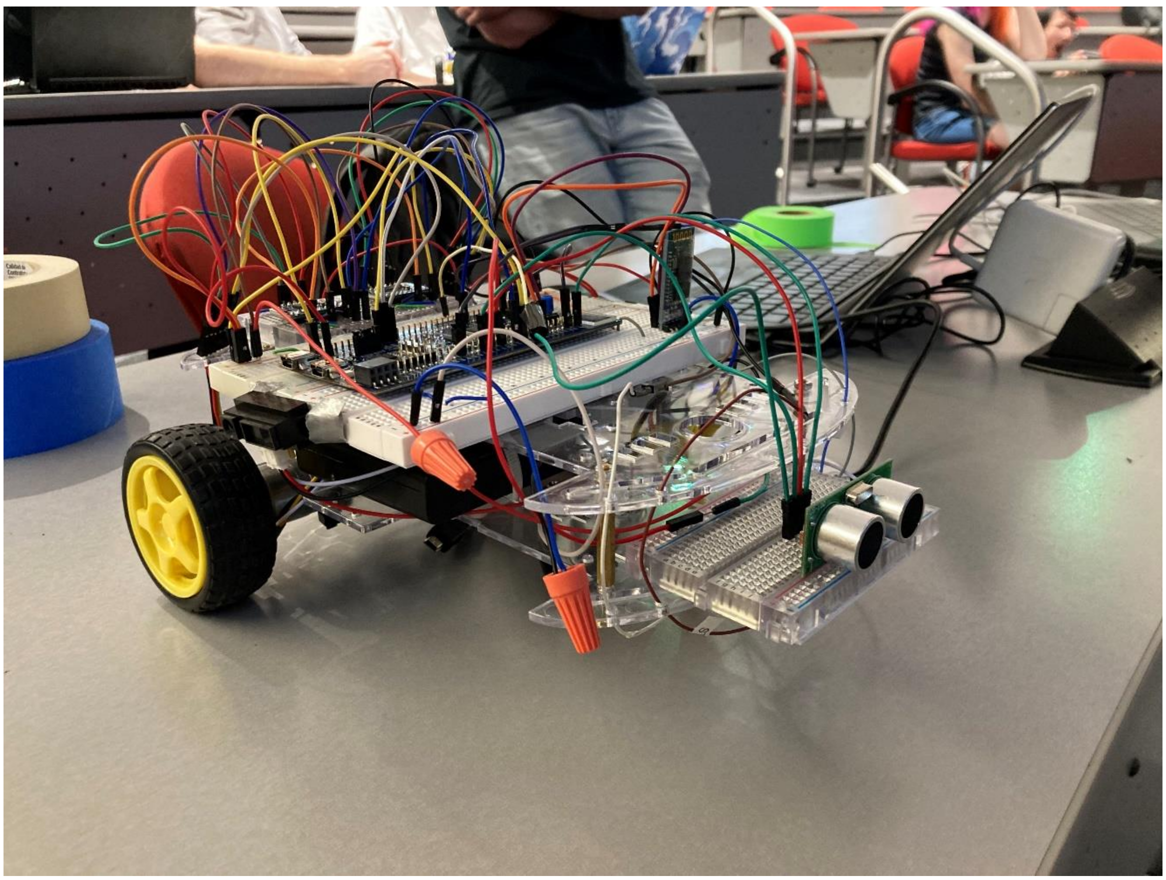

Fig. 1 – Assembled Rover: IR sensor can be seen on side duct taped to the bread board and the ultrasound sensor can be seen hanging out in the front.

Bill of Materials:

Item	**Quantity**	**Price Per Unit**	**Link**
20 mm Caster Bearing wheel	1	1.95	https://www.adafruit.com/product/3948
DC Gearbox Motor	2	14.88	https://www.amazon.com/dp/B07GNGQ24C?psc=1&ref=ppx_yo2ov_dt_b_product_details
Plastic Chassis	1	18.99	https://www.amazon.com/gp/product/B06VTP8XBQ/ref=ppx_yo_dt_b_asin_title_o07_s00?ie=UTF8&psc=1
8 AA Battery Holder (12V)	1	8.00	https://www.amazon.com/gp/product/B00VE7HBMS/ref=ppx_yo_dt_b_asin_title_o05_s00?ie=UTF8&psc=1
4 AA Battery pack (6V)	1	5.99	https://www.amazon.com/Duracell-CopperTop-Batteries-All-Purpose-Household/dp/B00000JHQ6/ref=asc_df_B00000JHQ6/?tag=hyprod-20&linkCode=df0&hvadid=309789297054&hvpos=&hvnetw=g&hvrand=6607104218528133970&hvpone=&hvptwo=&hvqmt=&hvdev=c&hvdvcmdl=&hvlocint=&hvlocphy=9029755&hvtargid=pla-637193815171&psc=1
IR Sensor	1	14.95	https://www.adafruit.com/product/164
Ultrasonic Distance Sensor	1	13.90	https://www.adafruit.com/product/4019
Color Sensor	1	7.95	https://www.adafruit.com/product/1356
Wheels	2	3.00	https://www.adafruit.com/product/3763
STM320F Discovery board	1	N/A	Already purchased in class
Motor Drivers Plan A: Our group's Motor Drivers Plan B: Commercial	3	Plan B: 6.95	Plan B: https://www.adafruit.com/product/2448
Breadboards	1-3	~ 5	N/A: (purchased from University stock room)
Wires/Solder /Misc.	N/A	~ 5	N/A: (purchased from University stock room)
Total:		**$150**	(Approximate value depending on component substitutions)

(Block Diagram is included at the end of the proposal)

Project Schedule:

Milestone 1 Goal (Mar 17): At this stage we will have ordered all of our parts and have come up with a schematic based off of our components' datasheets. We will also have ordered any backup/contingency parts, purchased necessary parts from the stockroom, and ordered any last minute parts we forgot about during planning stages. And finally we will have chosen the pins on the STM32F discovery board that we will be using and identified which register values we think we need to modify for correct function.

> Milestone 1 Actual: At this stage we were able to order all our initial parts and we had ordered most of our back up parts. We were able to select some of the final pins but these had to updated as we went. For example we had to change our USART and Motor driver pins.

Milestone 2 Goal (Mar 24): At this stage we will have assembled our board or in the worst case have the layout ready so we just need to insert any parts that haven't arrived yet. We also will have done some initial testing with our different components to see if replacement parts need to be purchased.

> Milestone 2 Actual: At this stage our board was not completely assembled but we had been able to verify that our ultrasound sensor was working and that the IR sensors were outputting a voltage.

Milestone 3 (Mar 31): At this stage we will have figured out how to communicate with each of the sensors and made sure that we have figured out the code necessary to communicate with the discovery board.

> Milestone 3 Actual: At this stage some back up components still hadn't arrived but we were able to talk with the color sensor, the ultrasound sensor, and the IR sensor. We were still waiting to receive our Bluetooth USART module but had started on the code.

Milestone 4 (Apr 7): At this stage we will have figured out how to operate the rover and have tested our obstacle detection abilities on a variety of obstacles.

> Milestone 4 Actual: At this stage our IR sensors were giving us grief but we had figured out our color sensor code and were able to successfully complete the drag race. Our IR code, motor code, and USART code seemed to be overlapping. We were also able to get the Bluetooth USART connection working and communicating with our computer.

Milestone 5 (Apr 14): At this stage we will have tested our drag race ability and finalized our obstacle detection abilities and have the rover at a basic working level.

> Milestone 5 Actual: At this stage the drag race code was still working but we still were having issues with the IR sensors. We weren't able to get read outs at this point and the ultrasound sensor stopped working properly. Eventually we figured out that we were having issues with our initializations overlapping.

Milestone 6 (Apr 21): At this stage our goal is to have optimized our approaches for both events and have our best solution finalized.

Milestone 6 Actual: At this stage we figured out the ultrasound and IR sensor overlapping errors but we still weren't getting good readouts from both IR sensors. We decided that we would just leave one IR sensor on there and use it to hug one of the walls while trying to get through the obstacle course.

Risks, Unknowns, and Potential Problems:

1. We do not know how sensitive any of our sensors will be in practice and will have to trust the advertised values for right now.
    1. Plan: Assume that the sensors will not work perfectly and underestimate their ranges
    2. Result: Our sensors' sensitivity didn't end up being an issue. We were able to rely on the values that were given to us from the sensors.
2. In addition to not knowing the sensitivity, we are not sure how well the color sensor will pick up the line for the drag race and how well the proximity sensors will detect the obstacles.
    1. Plan: Test the sensors early on with a range of obstacles/lines to make sure that the sensors work as planned and have back up sensors selected
    2. Result: In the end our tests for the color sensor picking up the line went well but in the actual event it stopped working. We were surprised because all tests beforehand showed really good results. We also found that as long as our obstacles were tall enough we were able to detect them and didn't see much of a difference between materials.
3. We also do not have extensive collective experience in this type of project and will be working with difficult topics without having experience outside of the lab.
    1. Plan: Ask for help and review lab handouts thoroughly throughout coding process
    2. Result: The lab handouts were helpful along with the labs. We tried to reuse our lab code as much as possible to stick to topics we were familiar with which helped out in the end.
4. We have not yet received our motor driver boards and we do not know if any of our boards will be operational.
    1. Plan: If our boards aren't working early on order commercial options quickly and start testing them/working with them early on
    2. Result: We ended up having to order new driver boards.
5. In addition, if we need to purchase new motor driver boards we will need to learn how to have them work with the discovery board
    1. Plan: Order commercial boards as a back up and test them out early on to see if they will work
    2. Result: The new boards were really easy to use since they had the same underlying IC.
6. Parts could take longer to arrive than expected or mistakes in the orders could occur.

    1. Plan: Check with the stock room or other local hobby shops to see if they have similar parts
    2. Result: Parts came quickly enough on Amazon that this wasn't an issue.
7. Software makes a few assumptions that a black line will be used to mark the finish line of both the obstacle course and drag race. That the obstacles will be largely spaced(0.5 meters apart), not navigating a maze.
    1. adjustments will be made as more details are disclosed.
    2. Result: We ended up trying to tune the color sensor for a blue line instead of a black line but this didn't seem to cause much of an issue. And obstacles were placed far enough apart for us to get our rover through. We had an issue with angled obstacles; our ultrasound sensors couldn't accurately measure the angled boxes due to the sound pulses not being received correctly along with facing a wall at an angle.

Software Block Diagram:

As mentioned earlier, at first we planned on having an autonomous hierarchy (see Fig. 2 below) but had to switch to manual due to time constraints and sensor issues. In the end we had a simple architecture where we would move and then pull sensor data and would make decisions to turn left, turn right, or go straight based on the received data.

Takeaways/Conclusion:

In the end we were able to make a functioning rover that was able to complete both tasks with relative success. Tests before the drag race were completed successfully despite the code not working during the actual race (our threshold levels seemed to react differently so an incorrect value was selected to trigger a stop). We also were able to get through the obstacle course while only making a few collisions with angled boxes.

The rover could be improved dramatically still. Namely through adding additional sensors (to account for blind spots on the rover and blind angles); adding autonomous structure (again see Fig. 2); and potentially creating a custom PCB board to improve spotty wire connections. In more minute detail it would have been good to consolidate or modulate our code better as well because we often ran into overlapping initializations of registers that would impair functionality. A single function that initialized everything would have been preferred.

Additional Documents/Figures:

Included below in Fig. 3 is an electrical wiring diagram and Fig. 4 is a Pin mapping document displaying the different pin selections.

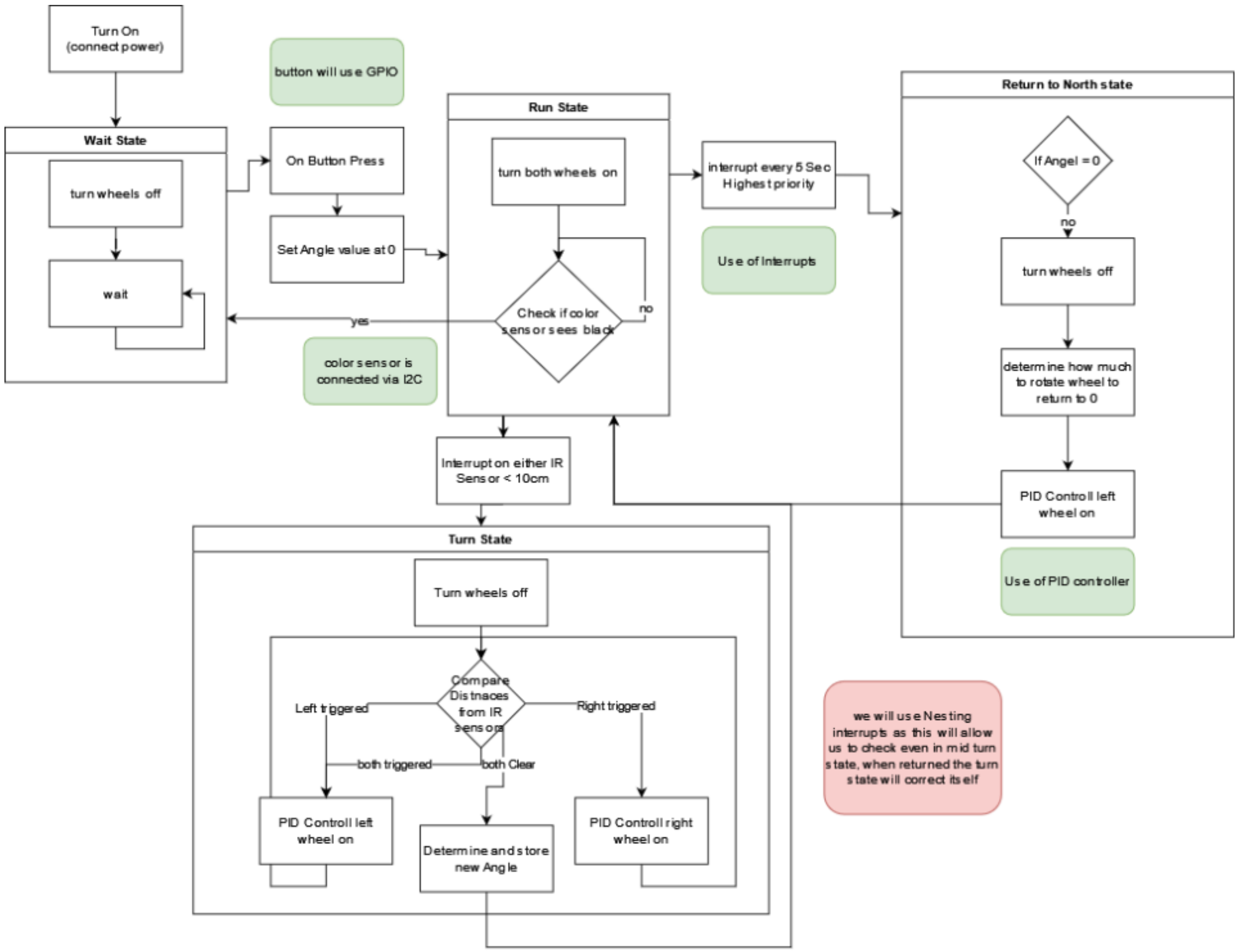

Fig. 2 Original software plan- Initial goal was to use interrupts and autonomy but in the end a manual drive was selected along with polling sensor data.

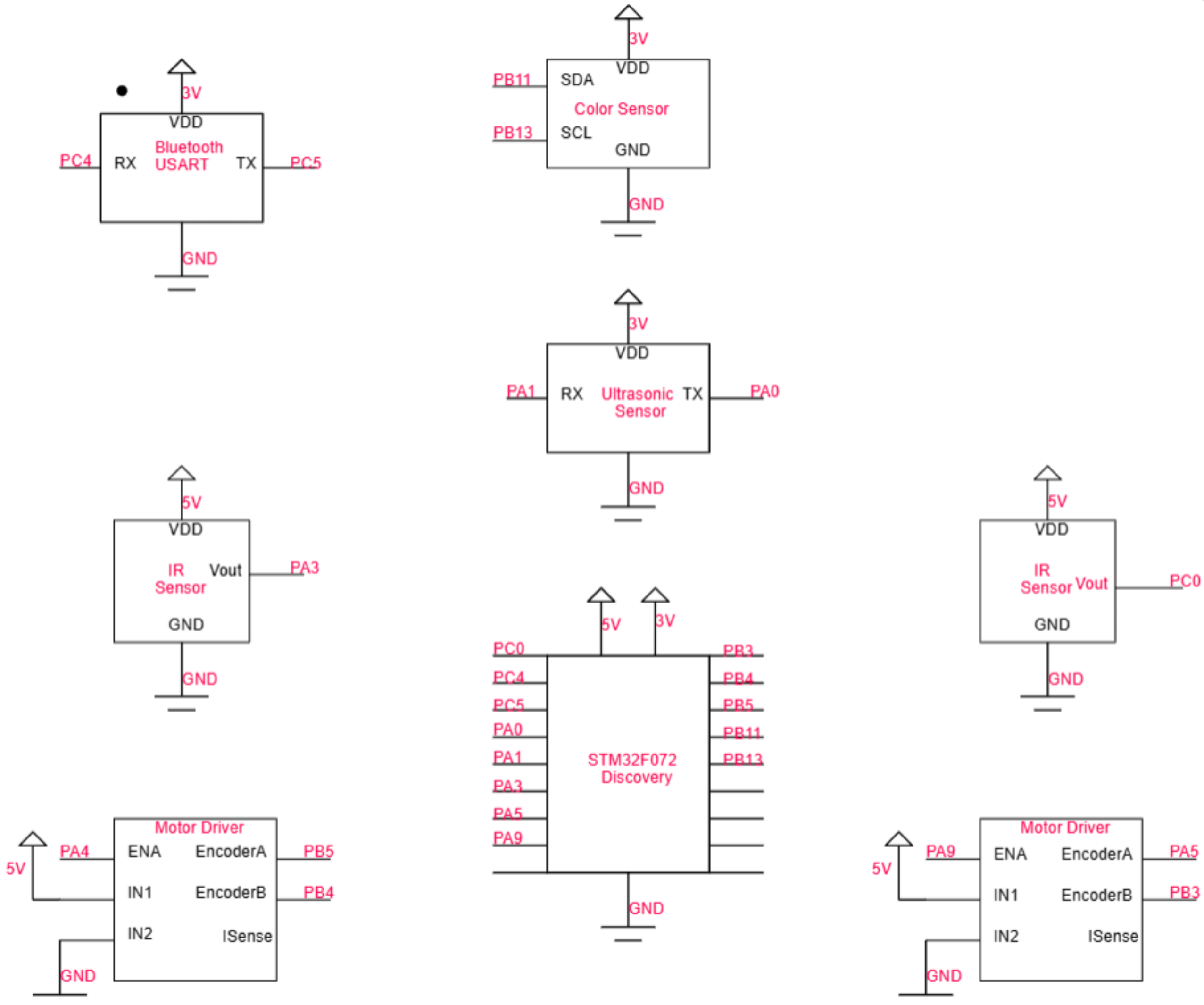

Fig. 3 Electrical wiring diagram

Sensor Pins	**STM32F072 Pins**
Color Sensor - VDD	3V
Color Sensor - GND	GND
Color Sensor - SDA	PB11 (AF 1 for I2C2 - SDA)
Color Sensor - SCL	PB13 (AF 5 for I2C2 - SCL)
IR Sensor1 - VDD	5V
IR Sensor1 - GND	GND
IR Sensor1 - Vout	PC0 (Analog Mode)
IR Sensor2 - VDD	5V
IR Sensor2 - GND	GND
IR Sensor2 - Vout	PA3 (Analog Mode)
Ultrasound Sensor - VDD	5V
Ultrasound Sensor - TX	PA0 (AF 4 for USART4 TX)
Ultrasound Sensor - RX	PA1 (AF 4 for USART4 RX)
Motor Driver1 - ENCODERA	PB5 (AF1 for TIM3 Ch2)
Motor Driver1 - ENCODERB	PB4 (AF1 for TIM3 Ch1)
Motor Driver1 - ENA	PA4 (AF 4 for TIM14 Ch1)
Motor Driver1 - IN2	3V
Motor Driver1 - IN1	GND
Motor Driver2 - ENCODERA	PA5 (AF1 for TIM2 Ch1)
Motor Driver2- ENCODERB	PB3(AF1 for TIM3 Ch2)
Motor Driver2- ENA	PA9 (AF2 for TIM 1 Ch1)
Motor Driver2- IN2	GND
Motor Driver2- IN1	3V
Bluetooth - TX	PC5 (AF1 for USART3 RX)
Bluetooth - RX	PC4 (AF1 for USART3 TX)
Bluetooth - VCC	5V

Bluetooth - GND	GND

Fig. 4- Table of pin assignments used on the STM32F0 discovery board