ROBOT_NAME

HERE_THE_SHORT_DESCRIPTION

PROJECT SPRINT #2.
DATE: 03 May 2017

Daniel Azemar (1390451)
Sergi Pous (1390450)
Jialuo Chen (1390203)
Adrià Rico (1391488)

Table of Contents

ROBOT_NAME

ROBOT_NAME

ROBOT_NAME

HERE_THE_SHORT_DESCRIPTION

Project description

The sole purpose of RAT is to transport an object from one point to another. To accomplish this, RAT will wait until it remotely receives a coordinate to go to, then it'll travel there while avoiding any obstacle in the way. It'll also periodically send its current position to the remote controller, so that the robot can be localized at any time.

Below is a detailed description on how we plan to carry out these functionalities:

- Movement:
  RAT will be able to move using two DC motors (with wheels attached) controlled by a motor driver board. This setup makes it possible for RAT to move forwards and backwards with an adjustable speed. Rotation will be accomplished by activating only one of the two motors.

- Remotely sending and receiving data:
  We'll use two transceivers controlled by two Arduino UNO boards.
  One of them will be located in the robot, and the other will be connected to the computer used to send the coordinates and receive the current robot position.

- Route tracing:
  When RAT receives an objective coordinate, it'll get its current position and orientation using a GPS and a magnetometer respectively. It'll then trace a straight line from its current position to the objective coordinate, calculate the direction, and rotate accordingly. When it determines it's facing the correct direction, it'll start moving forward until an obstacle is found or the objective coordinates have been reached. Periodical checks will be made to make sure it's not going offtrack.

- Obstacle avoiding:
  RAT will be able to detect nearby obstacles using a sonar. Once an obstacle is detected, it'll try to overcome it and then resume its original path. We still don't know a lot about how to code this, but we've been promised we're going to be taught.

Note on the limitations of the distance traveled:

The precision of the GPS used is around 3m, so RAT's duty is to land inside an area of 3m around the specified point. Specifying points too close to RAT would make it
The transceiver used has a range of about 100m, so making RAT go towards points farther than that leads to losing all kind of communication with it. It could still, in principle, autonomously get to its objective, and then receive new orders from a different computer nearby.
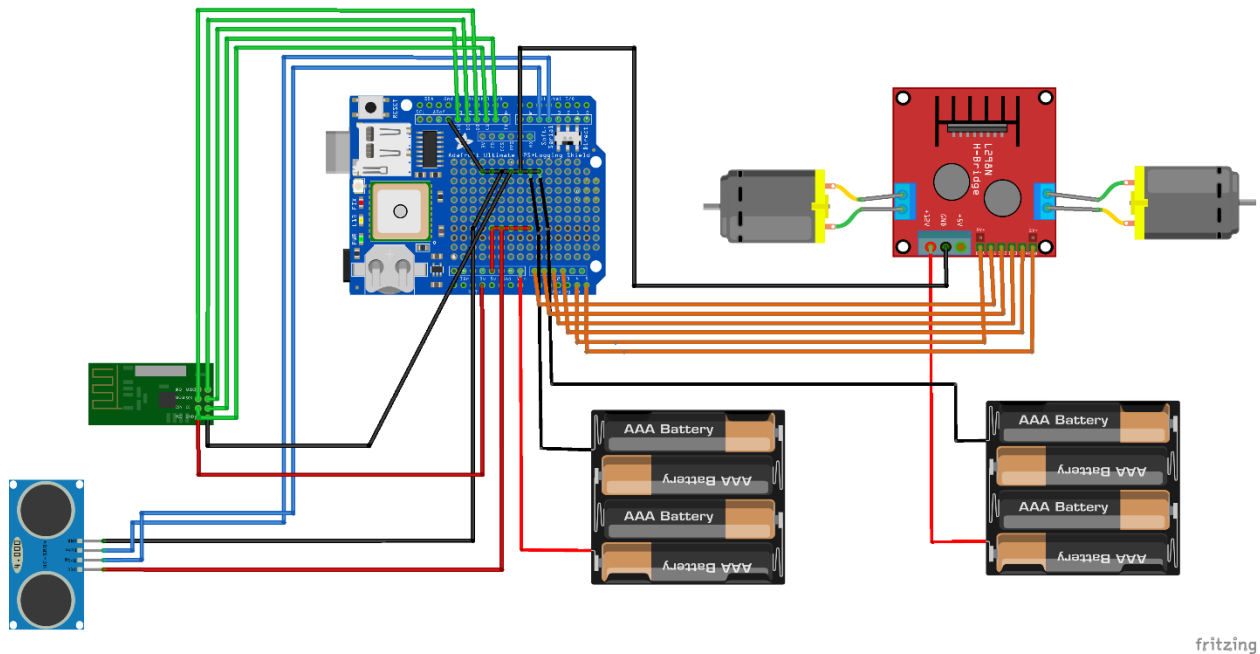
Electronic components

- Arduino UNO (2x) (one of them is of our own, it'll be used to send data from the computer to the robot using the transceiver)

- nRF24L01P+ (Transceiver) (2x)

- L293D (Motor Driver Board)

- HC-SR04 (Sonar)

- Adafruit Ultimate GPS Logger Shield

- DC motor (2x)

- Magnetometer (not yet received, it'll not be taken into account for the connections scheme below since we don't know the model)

Scheme

Notes:

- All red wires go to Vcc (5V for all the components except the transceiver which goes to the 3.3V pin).

- All black wires go to GND.

- All the AAA batteries shown in the scheme are 1.2V AA batteries in actuality.

- The GPS shield uses digital pins 0, 1 (for the direct connection), 7 and 8 (for the soft serial connection).

Connections:

Sonar (blue)

Trigger Pin -> Digital Pin 5 (PWM)

Echo Pin -> Digital Pin 4

ROBOT_NAME

Motor Driver (orange)

ENA -> Analogic Pin A4 (Digital Pin 18)

ENB -> Analogic Pin A5 (Digital Pin 19)

IN1 -> Analogic Pin A0 (Digital Pin 15)

IN2 -> Analogic Pin A1 (Digital Pin 15)

IN3 -> Analogic Pin A2 (Digital Pin 16)

IN4 -> Analogic Pin A3 (Digital Pin 17)

Transceiver (green)

CE Pin -> Digital Pin 9 (PWM)

CSN Pin -> Digital Pin 10 (PWM)

SCK Pin -> Digital Pin 13
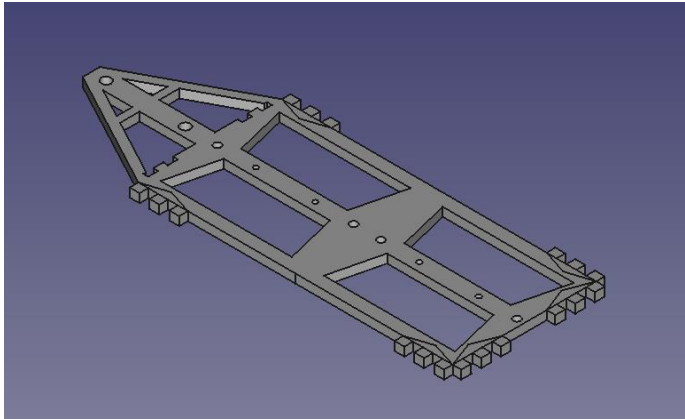
MOSI Pin -> Digital Pin 11 (PWM)
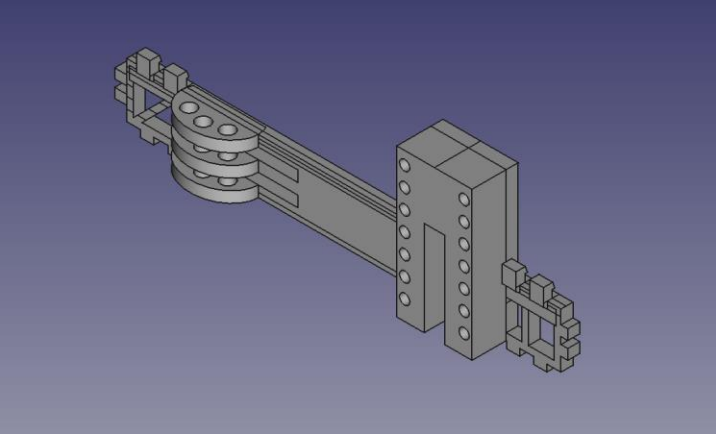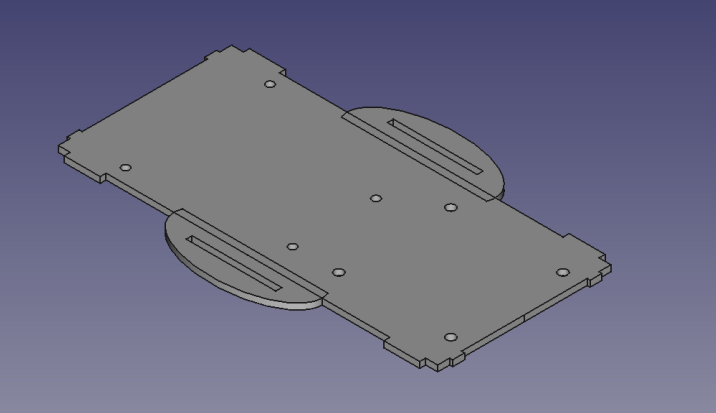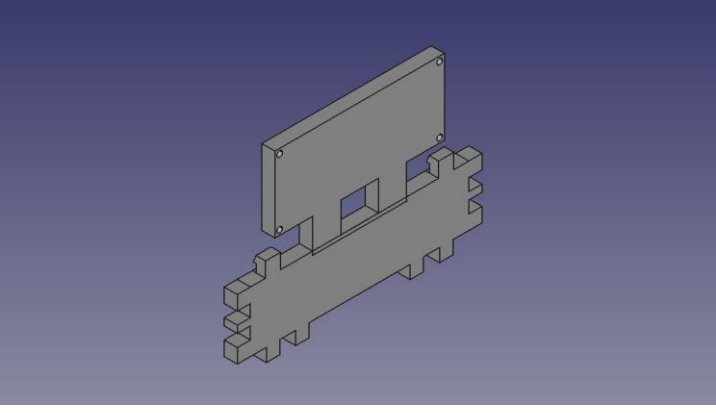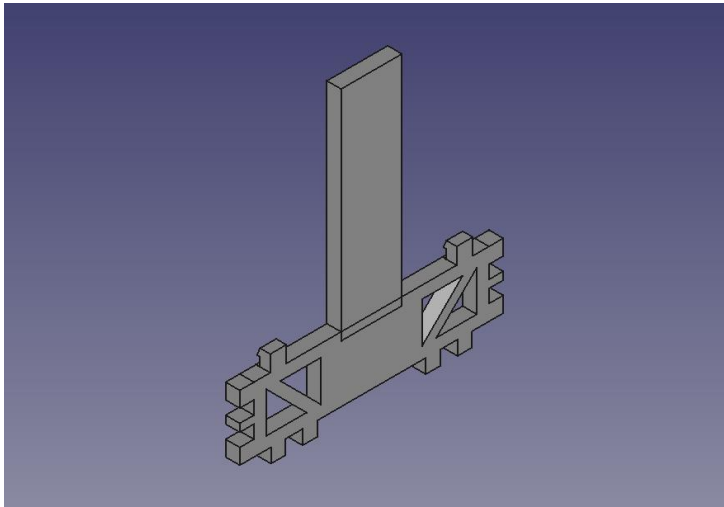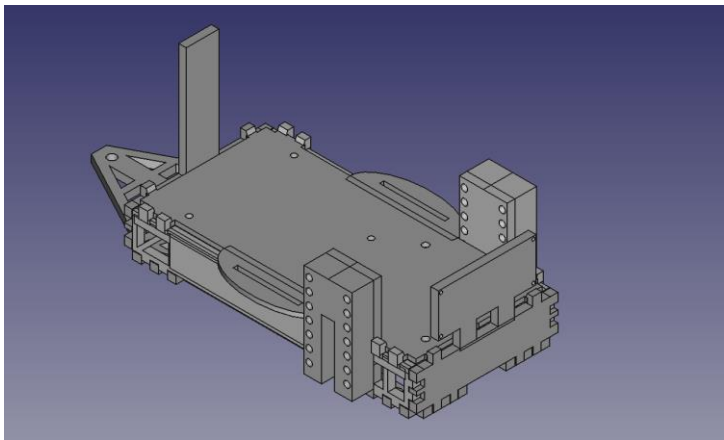
MISO Pin -> Digital Pin 12

Extra components and 3D pieces

- **Wheel for DC motor (2x)**
  They'll be connected to the motors to allow RAT to move on ground.

- **Caster wheel**
  A support wheel which will be placed on the back of the robot.

- **4 AA Battery Holder (2x)**
  Used to conveniently hold 4 AA batteries in series so that they produce 5V of voltage.

- **1.2V AA Rechargeable Battery (8x)**
  4 of them them are to power the Arduino board, the other 4 to power the motors.

List of 3D pieces:

No.	Description	Image
1	**Base** of the robot. Here we'll attach both the Battery Holder and the Caster Wheel.	

2	**Laterals** of the robot. Here we'll attach the motors. ...and a mystery piece, coming up next sprint!	
3	**Cover** of the robot. Here we'll attach both the Arduino UNO board and the Motor Driver Board.	
4	**Front** of the robot. Here we'll attach the Sonar.	

5	**Back** of the robot. Here we'll attach the Transceiver.	
ALL	Overall view of the structure.	

Note:

We'll join all the pieces together by sticking their respective hooks and ledges with superglue. However, the cover will use Snap-fit to make it easy to put it in and out, since the Arduino Board and the batteries are inside the robot, and we'll want to have easy access to them.

Foreseen risks and contingency plan

No.	Description	Probability	Impact	Contingency plan
1	We don't receive the magnetometer	Low	High	Use trial and error to determine the current orientation.
2	The magnetometer hasn't got enough precision	Medium	Medium	Direction checks should be done very frequently (stopping the robot).
3	The motors are not firmly fixed (and they make the robot tilt to the right or left instead of going in a straight line)	Medium	Medium	Redesign motor attachment to the robot and/or use superglue.
4	Misalignment or rupture of the snap-fit joints (between the cover and lateral 3D pieces).	High	Low	1. Redesign the joints and reprint the pieces. 2. (In case 1 keeps failing) Design and print a hinge which will be used to attach the cover with the laterals. 3. (In case 2 were to fail too) Just come up with a different way of accessing the insides of the robot.
5	Misalignment or rupture of the joints to be sticked with superglue.	Medium	Medium	Redesign the joints and reprint the pieces.

References

This project has been inspired by the following Internet projects:

http://www.instructables.com/id/Arduino-Powered-Autonomous-Vehicle/

http://www.robotshop.com/letsmakerobots/fundamentals-a-gps-guided-vehicle