**2020 ME Senior Project Description**

**& To-do List**

**Instructor: Michael Lu, Senbao Lin**

**Group 1 member:**

**Keith Chen**

**Larry Jin**

**Sherlock He**

**Hugo Li**

**Group 2 member:**

**Hai Li**

**Oliver Yu**

**Elazar Pan**

**Adonis Guo**

|  |  |
| --- | --- |
| **Update Note** | |
| **9.10** | **Build the document** |
| **9.12** | **Add the full group member name and requirement for each program** |
|  |  |
|  |  |

9.10

Robot description: A bomb disposal robot mounted on the McNamee wheel chassis.

Function: all direction remote control movement, through visual recognition command arm cut "red line on the bomb"

Composition:

1. Chassis consisting of four McNamee wheels,

2. Dobot robot arm as the main action module & remote control receiving end.

3. Equipped with raspberry PI and camera as upper computer for visual recognition (alternative: Openmv)(note: 1, 2 are responsible for Group 1 and 3 are responsible for Group2)

To-do list:

Group 1:

1. Design and built the chassis (the chassis should be reserved for the position of motor bracket, battery, chassis core plate (arduino tentatively) and raspberry pie space (heat dissipation), and the position of cylinder)
2. Investigated and purchased the motor & electric adjustment (Chen guanbo), and customized the motor bracket
3. Investigate whether the remote control part of Dobot can be modified
4. Control code rewriting

Group 2:

1. Be familiar with raspberry pie

2. Install and understand opencv on raspberry pie

3. Design and installation of camera bracket4. Complete the required functions.

Division of labor:

Team leader: Keith Chen.

Group 1:

Chassis design, processing and assembly: Larry Jin, Sherlock He

Code for chassis: Hugo Li

Code for remoter & control task: Keith Chen

Group 2:

Robot Arm motion design & test: Adonis Guo, Oliver Yu

Vision algorithm design: Hai Li, Elazar Pan

9.12

**Requirement for each program**

Chassis:

1. Using PID control to control the wheels
2. Using outer interrupt to get the information from the encoder
3. Using timer interrupt to make the speed of the chassis increase slowly
4. Using direction solving algorithm to solve how much current should be sent to the motor
5. Reserve interface for remoter

Mission complete standard:

The chassis should have the ability to move forward for 5 sec, backward for 5 sec, leftward and rightward translation for 5 sec, Left & Right rotate for 5 sec. Through forward and backward process, the speed of the chassis should increase slowly.

Remoter & control task：

1. Reserve interface for the vision information
2. Use timer interrupt to let Arduino& Dobot receive the remote information
3. To avoid interrupt between remoter and Ruspberry Pi, the remoter only controls the Yaw axis of Dobot.
4. Make the order from the remoter can be done simultaneously

Mission complete standard:

Without chassis, the remoter should make the Dobot turn according to Yaw axis

With chassis, the robot should move as the order of the remoter

Vision:

1. Distinguish the color of the lines
2. Distinguish the center of the line and get the offset of it between the center of the image
3. Send offset to Arduino Mega 2560, and then, Mega 2560 will send it to the Dobot, make sure that the Dobot can cut the line.
4. Design the camera holder and scissor driven by air pump, assemble them to the Dobot.

Mission complete standard:

Finish the required task.