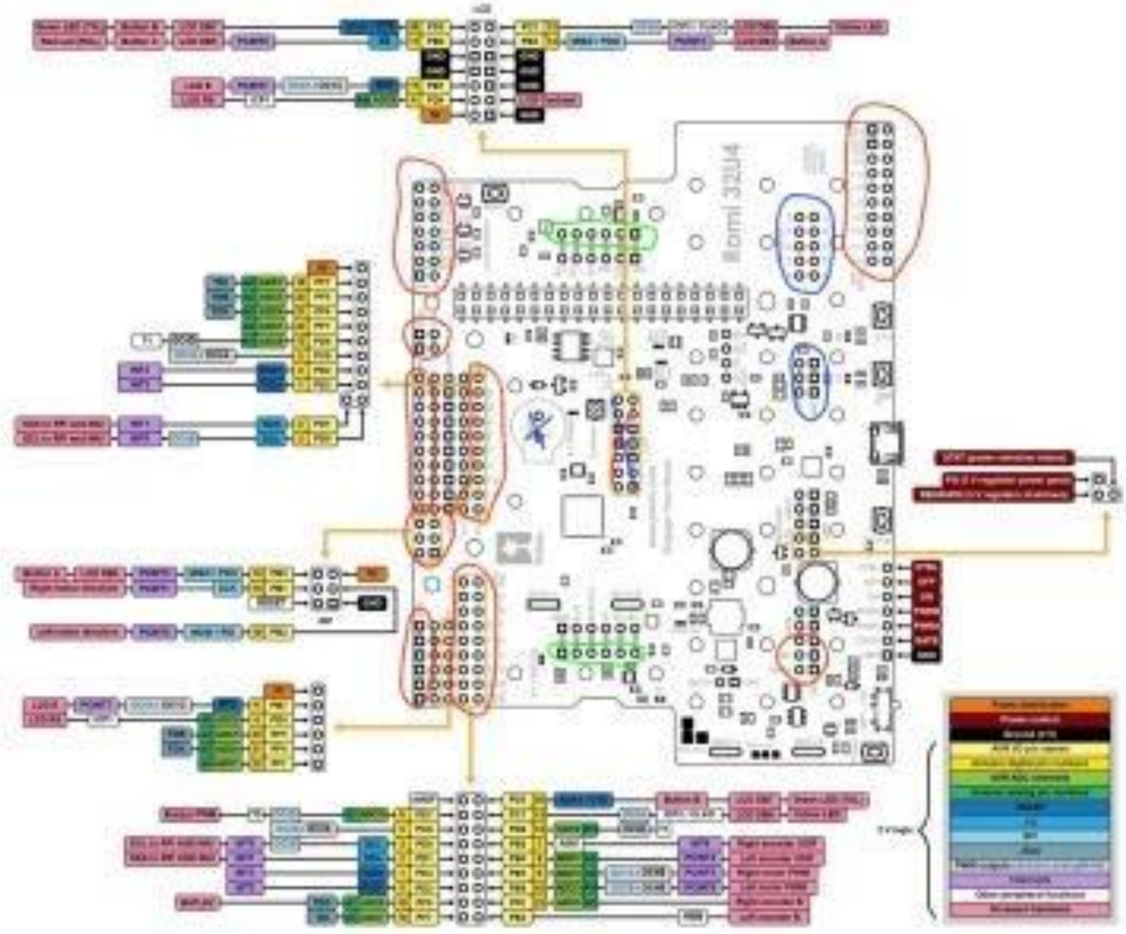
CPE476 – FALL 2020

Assignment 1

Student Name: Alex Contreras
Student #: 2001486950
Student Email: contra6@unlv.nevada.edu
Primary: https://github.com/alexcontreras7/mobile-robotics
Directory: Assignment 1

1. COMPONENTS LIST AND CONECTION BLOCK DIAGRAM w/ PINS



2. INITIAL/MODIFIED/DEVELOPED CODE (Code Provided)

https://github.com/venki666/CpE476_demos/tree/master/Arduino/Romi-NoRpi-Debug

3. DEVELOPED CODE

a) Develop the code to move the Romi Robot in a circle (you can use your own parameters)

```
#include
<Romi32U4.h>
               #include <PololuRPiSlave.h>


               Romi32U4Motors motors;
               Romi32U4Encoders encoders;
               Romi32U4ButtonA buttonA;


               void setup() {
                 Serial.begin(57600);


                 // put your setup code here, to run once:
                    buttonA.waitForButton(); //Wait for user input
                    delay(2000);

                 ledYellow(false);
                 ledGreen(true);
                 ledRed(false);
               }


               float _debug_linear_ms = 0.25;
               float _debug_angle_rs = 0.0;
               void _DEBUG_PID_CONTROL() {
                 static float _linear_ms_change = 0.1;
                 set_twist_target(_debug_linear_ms, _debug_angle_rs);
               }


               void loop() {


                 _DEBUG_PID_CONTROL();
                 // put your main code here, to run repeatedly:
                  _debug_angle_rs = 0.75; // radius
```

```
   set_twist_target(_debug_linear_ms, _debug_angle_rs); // sets the twist target

  if (everyNmillisec(10)) {
    // ODOMETRY
    calculateOdom();
    doPID();
  }


}
```

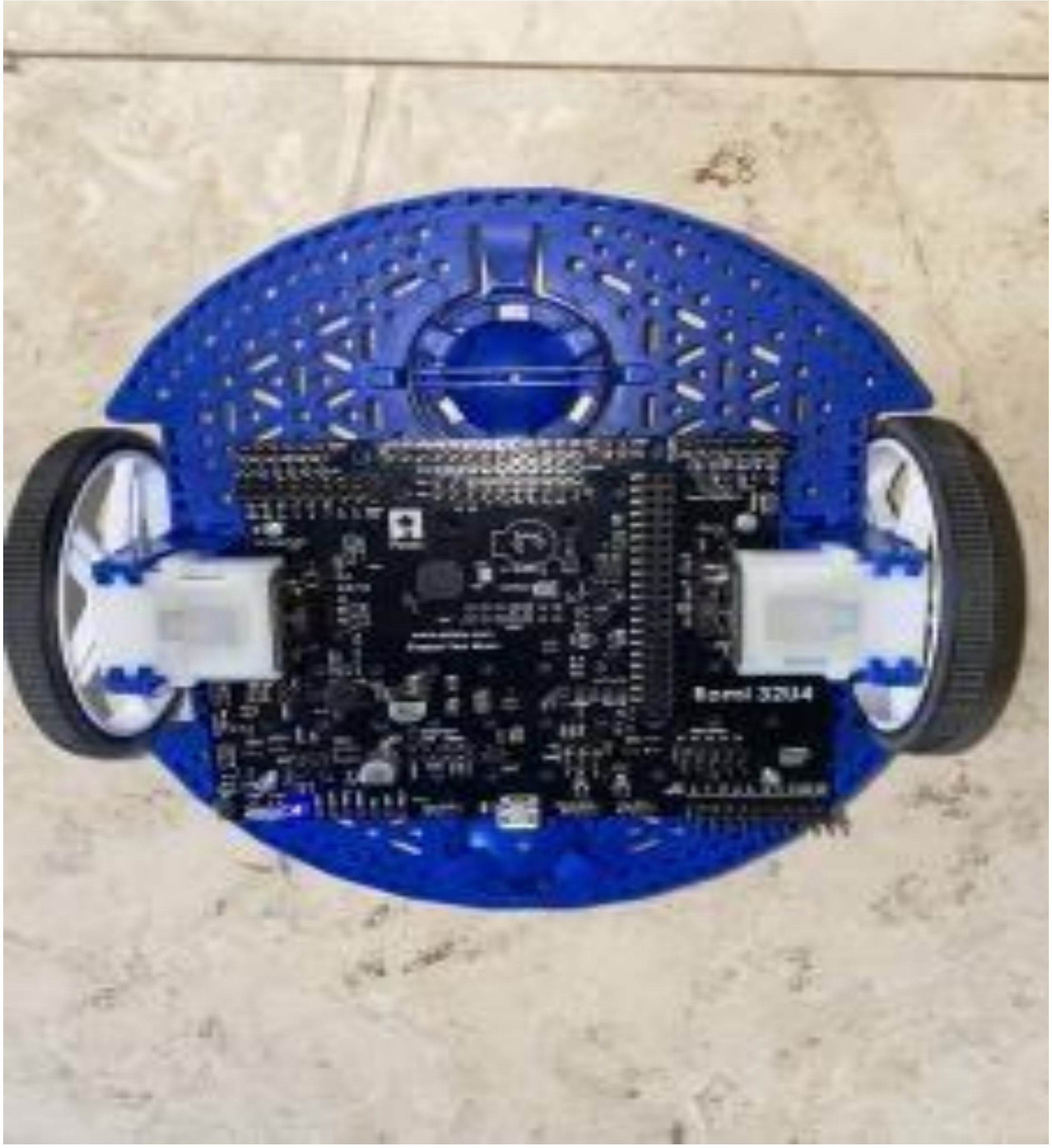
b) Develop the code to move the Romi Robot in a square/rectangle (you can use your own parameters)

```
#include
<Romi32U4.h>
               #include <PololuRPiSlave.h>


               Romi32U4Motors motors;
               Romi32U4Encoders encoders;
               Romi32U4ButtonA buttonA;


               void setup() {
                 Serial.begin(57600);


                 // put your setup code here, to run once:
                 buttonA.waitForButton(); //Wait for user input
                 delay(2000);

                 ledYellow(false);
                 ledGreen(true);
                 ledRed(false);
               }


               float _debug_linear_ms = 0.25;
               float _debug_angle_rs = 0.0;
               void _DEBUG_PID_CONTROL() {
                 static float _linear_ms_change = 0.1;
                 set_twist_target(_debug_linear_ms, _debug_angle_rs);
               }


               void loop() {

                 _DEBUG_PID_CONTROL();
                 float pi = 3.14159265358979323846; // value of pi
                 // for timing
                 unsigned int currentTime = millis();
                 unsigned int nextTime = currentTime + 2000; // delay
```

```
 while(currentTime < nextTime){   //go straight for 2000 ms
   if (everyNmillisec(10)) {
     // ODOMETRY
     calculateOdom();
     doPID();
   }
   currentTime = millis(); // updates time

 }


 // change directions
_debug_linear_ms = 0.0; // velocity
set_twist_target(_debug_linear_ms, _debug_angle_rs); // sets twist target


currentTime = millis();
nextTime = currentTime + 500; // delay
while(currentTime < nextTime){ //turn for half a second
   if (everyNmillisec(10)) {
     // ODOMETRY
     calculateOdom();
     doPID();
   }
   currentTime = millis(); // updates time
}




// change directions
_debug_angle_rs = pi * 0.5; // radius
set_twist_target(_debug_linear_ms, _debug_angle_rs); // sets twist target
currentTime = millis();
nextTime = currentTime + 1000; // delay
while(currentTime < nextTime){ //turn for a second
   if (everyNmillisec(10)) {
     // ODOMETRY
     calculateOdom();
     doPID();
   }
   currentTime = millis(); // updates time
}
```

```
// changes direction one last time
 _debug_angle_rs = 0.0; // raidus
 set_twist_target(_debug_linear_ms, _debug_angle_rs); // sets twist target
 currentTime = millis();
 nextTime = currentTime + 500; // delay
 while(currentTime < nextTime){ //turn for half a second
    if (everyNmillisec(10)) {
      // ODOMETRY
      calculateOdom();
      doPID();
    }
    currentTime = millis(); // updates time
 }

 _debug_linear_ms = 0.25; // velocity
 set_twist_target(_debug_linear_ms, _debug_angle_rs); // sets twist target
}
```

4. SCREENSHOT OF SETUP



5. VIDEO LINKS OF EACH DEMO

a) Circle Code:
https://www.youtube.com/watch?v=4cPH646BKlc&feature=youtu.be

b)
https://www.youtube.com/watch?v=eDUw4EH1Ssw&feature=youtu.be

6. GITHUB LINK OF THIS DA

https://github.com/alexcontreras7/mobile-robotics/tree/master/Assignment%201

Student Academic Misconduct Policy
http://studentconduct.unlv.edu/misconduct/policy.html

"*This assignment submission is my own, original work*".
Alex Contreras