Peter Huynh

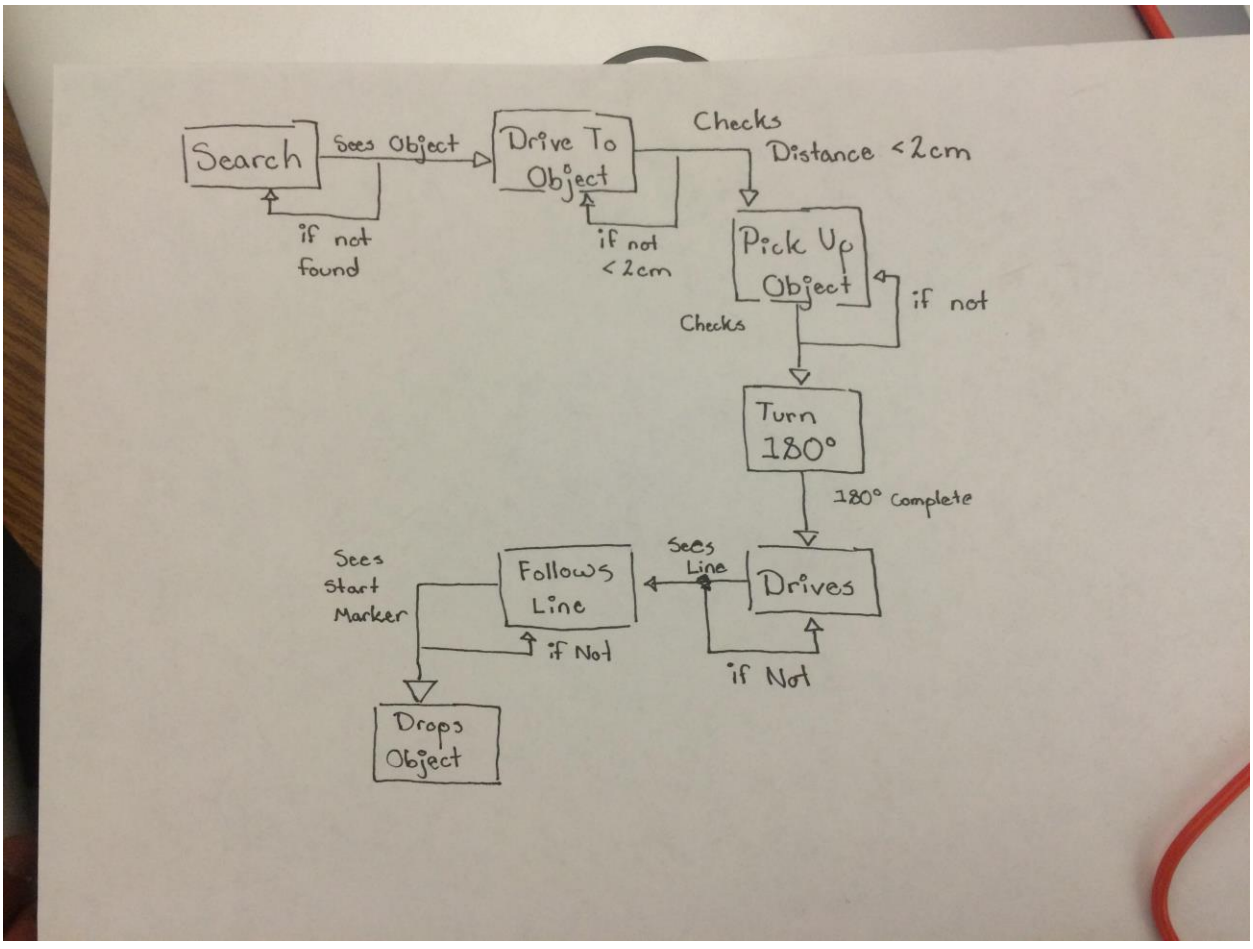
CSCI 3302

Intro to Robotics

September 4, 2016

Lab 1 Write-Up

Machine State Diagram:



Source Code:

```
#include <Sparki.h>

#define SEARCH 1
#define DRIVE 2
#define PICKUP 3
```

```
#define ROTATE 4
#define TOLINE 5
#define FOLLOWLINE 6
#define DROP 7

// Yolo
int stateId = SEARCH;

void setup() {
  // put your setup code here, to run once:
  sparki.servo(SERVO_CENTER); // center the servo
  sparki.gripperOpen();
  delay(3000);
  sparki.gripperStop();
}

void loop() {
  // put your main code here, to run repeatedly:
  int threshold = 900;
  bool endLoop = false;
  int cm, lineLeft, lineRight, lineCenter;

  if(stateId > 7 || stateId < 1)
  {
    sparki.moveStop();
    sparki.RGB(RGB_RED);
  }

  while(!endLoop)
  {
    switch(stateId)
    {
      case SEARCH:
        sparki.moveLeft(5);
        cm = sparki.ping();
        if(cm != -1)
        {
          if(cm < 70)
          {
            sparki.beep();
            sparki.moveLeft(6);
            stateId = DRIVE;
          }
        }
        delay(100);
        break;
```

```
      case DRIVE:
        sparki.moveForward(5);
        cm = sparki.ping();
        if(cm != -1)
        {
          if(cm < 7)
          {
            sparki.beep();
            sparki.moveForward(2);
            stateId = PICKUP;
          }
        }
        delay(100);
        break;

      case PICKUP:
        sparki.beep();
        sparki.gripperClose();
        delay(3000);
        sparki.gripperStop();
        stateId = ROTATE;
        break;

      case ROTATE:
        sparki.beep();
        sparki.moveLeft(180);
        stateId = TOLINE;
        break;

      case TOLINE:
        lineLeft   = sparki.lineLeft();   // measure the IR sensors
        lineCenter = sparki.lineCenter
        lineRight  = sparki.lineRight();
        if ( lineCenter < threshold && lineLeft < threshold &&
lineRight < threshold  )
        {
          sparki.beep();
          sparki.moveForward(4); // move forward
          sparki.moveRight(90);
          stateId = FOLLOWLINE;
        }
        else
        {
          if ( lineCenter < threshold )
          {
```

```
      sparki.moveForward(4); // move forward
      if ( lineCenter < threshold )
      {
        stateId = FOLLOWLINE;
      }
    }
    if (lineLeft < threshold)
    {
      sparki.moveLeft(45);
      if ( lineCenter < threshold )
      {
        stateId = FOLLOWLINE;
      }
    }
    if (lineLeft < threshold)
    {
      sparki.moveRight(45);
      if ( lineCenter < threshold )
      {
        stateId = FOLLOWLINE;
      }
    }
    else
    {
      sparki.moveForward(); // move forward
    }
  }

  sparki.clearLCD(); // wipe the screen

  sparki.print("Line Left: "); // show sensor on screen
  sparki.println(lineLeft);

  sparki.print("Line Center: ");
  sparki.println(lineCenter);

  sparki.print("Line Right: ");
  sparki.println(lineRight);

  sparki.updateLCD();
  delay(100); // wait 0.1 seconds
  break;

case FOLLOWLINE:
  lineLeft   = sparki.lineLeft();   // measure the sensor
  lineCenter = sparki.lineCenter();
```

```
    lineRight  = sparki.lineRight();

    if ( lineCenter < threshold )
    {
      if (lineLeft < threshold && lineRight < threshold)
      {
        stateId = DROP;
      }
      sparki.moveForward(); // move forward
    }
    else
    {
      if ( lineLeft < threshold )
      {
        sparki.moveLeft(); // turn left
      }

      else if ( lineRight < threshold )
      {
        sparki.moveRight(); // turn right
      }
    }

    sparki.clearLCD(); // wipe the screen

    sparki.print("Line Left: "); // show sensor on screen
    sparki.println(lineLeft);

    sparki.print("Line Center: ");
    sparki.println(lineCenter);

    sparki.print("Line Right: ");
    sparki.println(lineRight);

    sparki.updateLCD();

    delay(100); // wait 0.1 seconds
    break;

  case DROP:
    sparki.beep();
    sparki.moveStop();
    sparki.gripperOpen();
    delay(3000);
    sparki.gripperStop();
    endLoop = true;
```

```
        stateId = -1;
        break;

      default:
        endLoop = true;
        break;
  }
}
}
```

Summary:

Overall, this lab has been a great learning experience. We applied what we learned from lab 0 into action after we were given a set of specific tasks to accomplish. Initially, our team started off with a machine state diagram, in order to understand the basic structure of our simple switch statement robot code. After complete the diagram, we jumped into creating the switch diagram skeleton structure, and then applied what we saw within the wall avoidance, light follow, and line follow example codes into our switch cases. Once all the parts of code were in play, were adjusted aspects of the example codes to fit what we needed for the task at hand, and completed the lab.