

Khim Sok  
COMP.4510  
Mobile Robotics II  
Prof. Yanco  
Dec 14, 2021

### Final Project Video

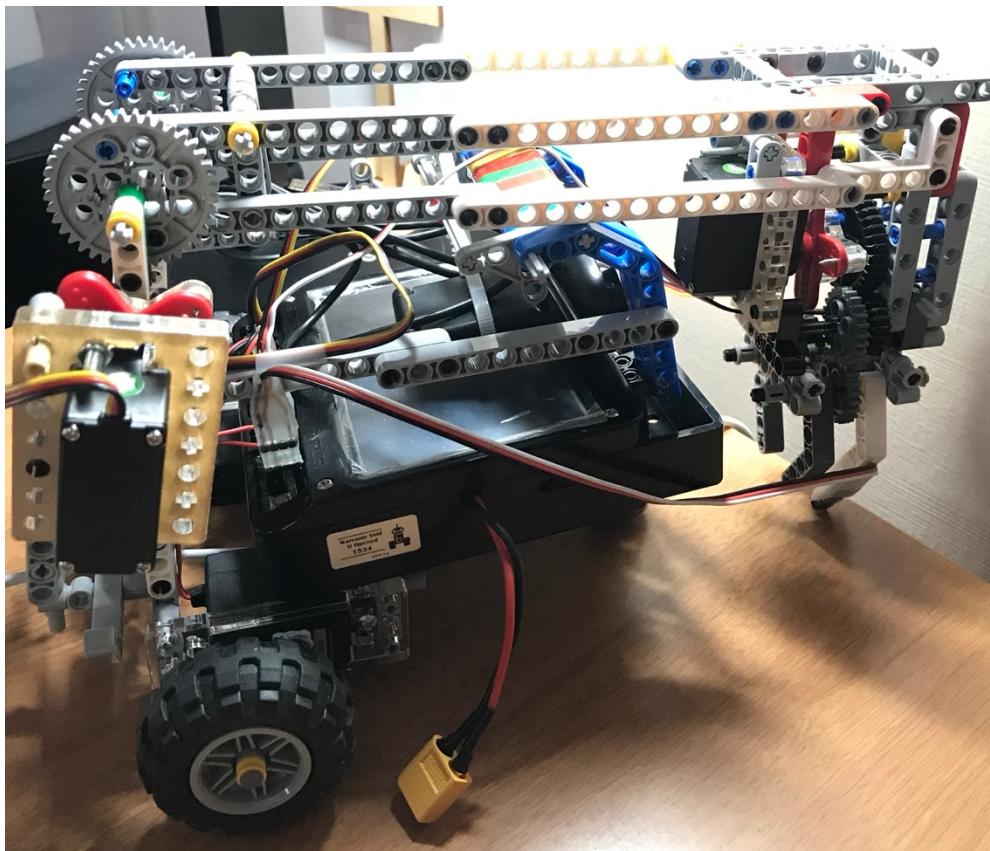
#### Video Link:

<https://drive.google.com/file/d/1vYCyrYuSVcm-MFltXkMtvOLijTLwzg3K/view?usp=sharing>

### Final Project Report

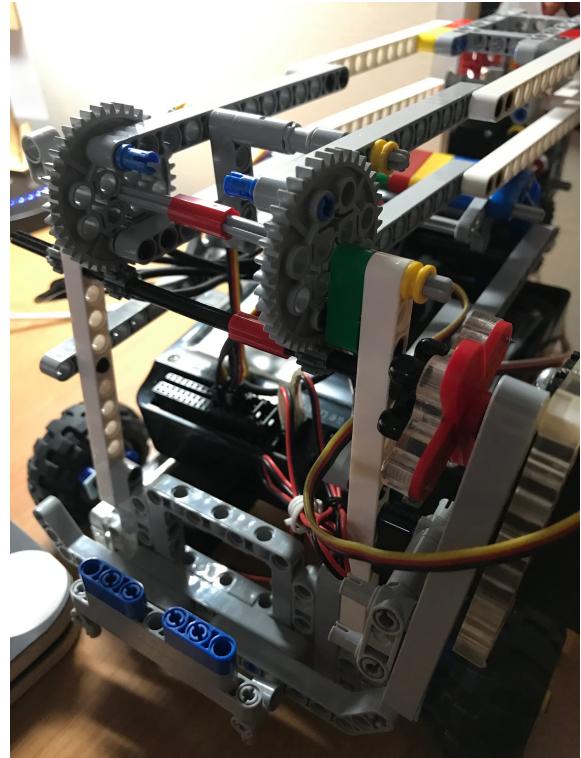
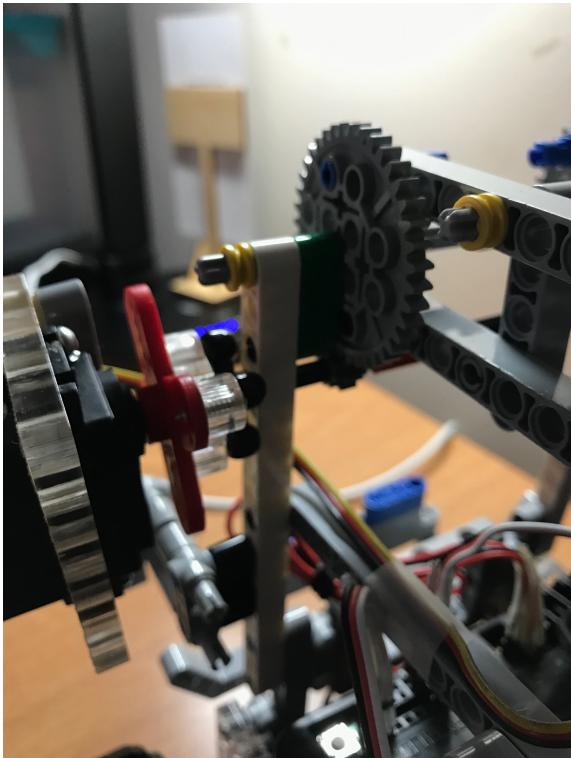
#### 1. Briefly Describe

The task of this robot is to use its arm and gripper to pick up color blocks. It should start by lifting its arm up to enable the camera to look for an available color block, once it found a block, it will identify the color immediately and then lower its arm to grab the block, if the robot detected that the block is not securely grabbed by the gripper, it will release the gripper and try to grab it again. If the robot detect that the block is successfully grabbed, it will lift up the arm and turn its body to either left or right depends on the color it picked up. After it find the spot for the color block, it will drop it slowly and release then turn back to its initial position. It will repeat the process until we stop the program.

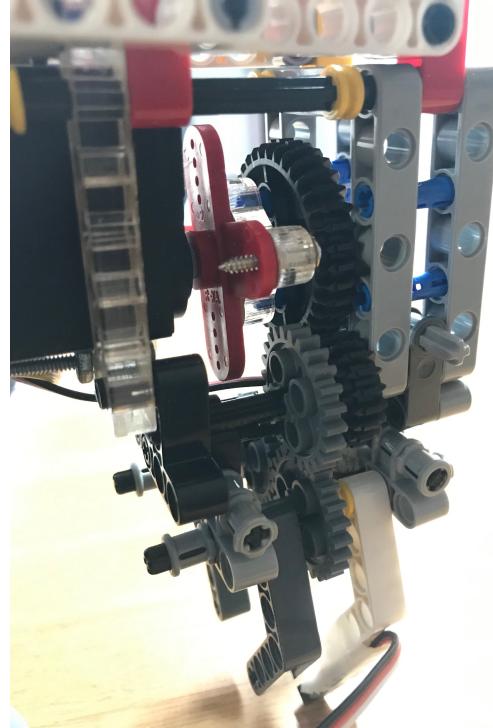


## 2. Design Structure

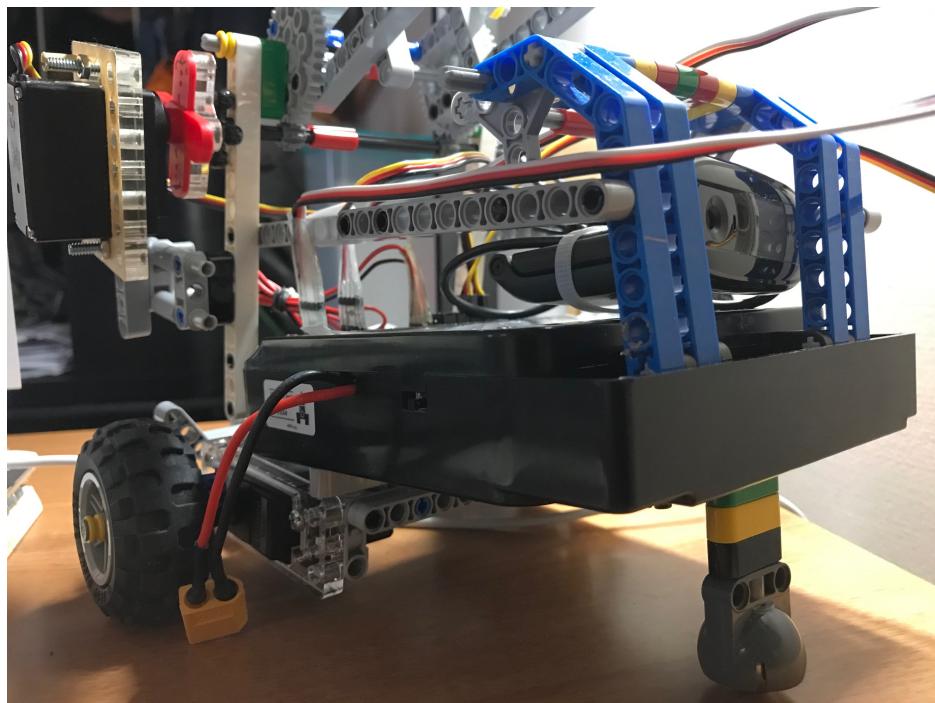
Shoulder: I use one servo motor to rotate two small gray 8 teeth gears and make it rotate the two other gray 40 teeth gear. This should be to lift up the arm up and down to its desire position.



Gripper: I use one servo motor attach with a black 40 teeth gears to rotate a smaller black gear with 20 teeth to also turn three other gray 24 teeth gears, together making a mechanic gearing system to open and close the gripper.

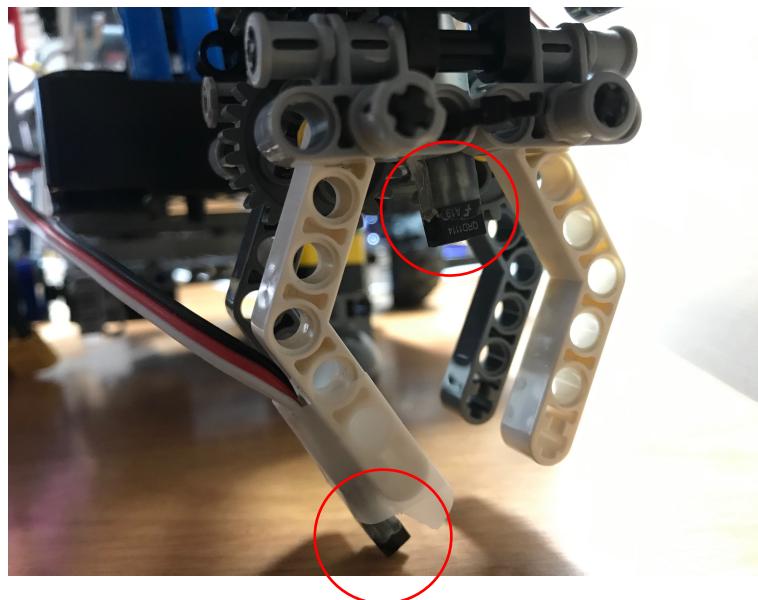


Base: I use the Wombat itself to attach with 2 motors to turn whole body left or right.

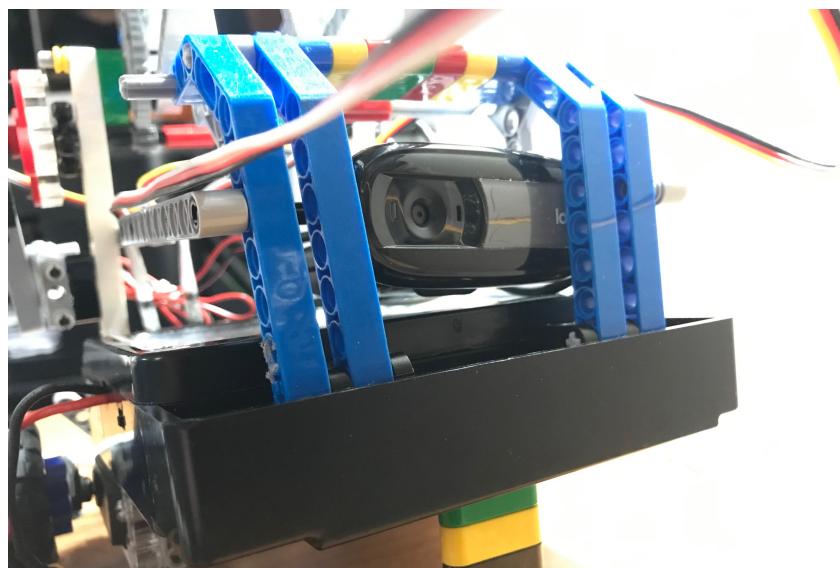


Sensor: I attach an IR reflectance distance sensor close to the gripper to detect whether the gripper is securely grabbing a block, if not, it will try again.

I attached another IR reflectance distance sensor to the edge of the gripper. Purpose of this is that when the robot tries to lower the arm to place the color block down the ground, if there is another block already exist on the ground where it supposed to place the block in its gripper, it will try to stack it on the other. If the IR sensor detect that it reaches the other block which mean it successfully stacked it. Then stop lower the arm, and it will release the block and back to its initial position.



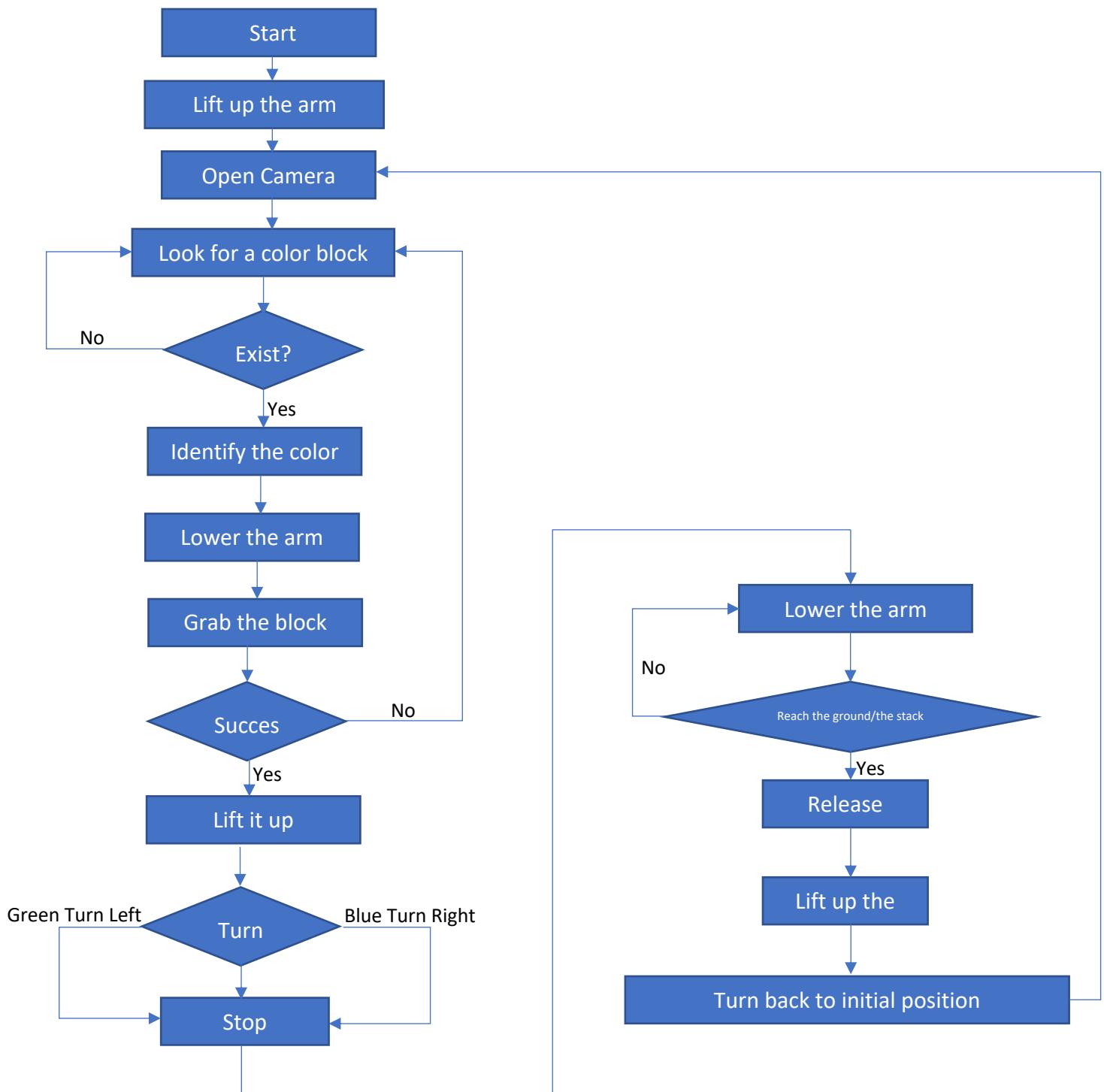
Camera: I use a color camera included in the kit to detect the color blocks since it has image processing included, it is very useful to identify color and the distance of each block.



### 3. Code Design

Behavior: Lift up the arm, look for a color block on the floor, if exist, identify the color, lower the arm, grab it, lift it up, if green turn left, if blue turn right. Drop it down, release, lift the arm up, back to initial position.

Flowchart:



#### 4. Source Code

```
#include <kipr/wombat.h>

void done_message(int small);
int is_done();

void take_it();
void put_down();
void join(int up_or_down);
void gripper(int close_or_open);
void find_cube();

void perform();

int main()
{
    perform();

    return 0;
}

void perform(){
    //repeat until user stop the program
    while(!is_done()){
        find_cube();
        msleep(500);
    }
}

void done_message(int small)
{
    //clears screen
    console_clear();
    if(!small) printf("Press \"C\" once finished.\n");
}

int is_done(){
    return c_button_clicked();
}

void take_it(){
    set_servo_position(0, 200);
    msleep(500);
    set_servo_position(0, 400);
    msleep(500);
    set_servo_position(0, 600);
    msleep(500);
    set_servo_position(0, 800);
    msleep(500);
    set_servo_position(0, 1000);
```

```

msleep(500);
set_servo_position(0, 1200);
msleep(500);
set_servo_position(0, 1450);
msleep(1000);
gripper(0);
msleep(500);
//if IR reflectance sensor detect the block is not securely grabbed,
//call perform function and repeat performance again. Otherwise, lift
it up
if(analog(0)>3100){
    perform();
}
msleep(1000);
set_servo_position(0, 10);
msleep(500);

}

void put_down(){

enable_servo(0);
int pos = 10;
printf("analog1 %d\n", analog(5));
//lower the arm until it reach the ground or the stack
while(pos<1600 && ((analog(5)>3000 && analog(5)<3480) ||
analog(5)>3400)){
    set_servo_position(0, pos+=100);
    printf("analog1 %d\n", analog(5));
    msleep(500);
}

//release
gripper(1);
msleep(500);
//lift it up
set_servo_position(0, 10);
msleep(1000);

}

//when this fuction is called, either lift up the arm or lower the arm
void join(int up_or_down){ //if input 0 put down, if 1 lift up

enable_servo(0);
set_servo_position(0, 2045);

int pos_servo = get_servo_position(0);

//put down slowly
if(up_or_down == 0){
    while(pos_servo < 2030){
        set_servo_position(0, pos_servo);
}
}
}

```

```

        pos_servo += 10;
        printf("servo0 is now at pos: %d\n", pos_servo);
        msleep(100);
    }
}

//lift up slowly
if(up_or_down == 1){
    while(pos_servo > 100){
        set_servo_position(0, pos_servo);
        pos_servo -= 100;
        printf("servo0 is now at pos: %d\n", pos_servo);
        msleep(100);
    }
}

msleep(1000);

disable_servo(0);
}

//if input 0 close, 1 open
void gripper(int close_or_open){

enable_servo(1);
int pos_servo = get_servo_position(1);

if(close_or_open == 0){
    set_servo_position(1, 1150);
    msleep(500);
    pos_servo = get_servo_position(1);
    printf("servo1 is now at pos: %d\n", pos_servo);
} else if(close_or_open == 1){
    set_servo_position(1, 1600);
    msleep(500);
    pos_servo = get_servo_position(1);
    printf("servo1 is now at pos: %d\n", pos_servo);
}

disable_servo(1);
}

void find_cube(){

enable_servo(0);

//lift up the arm
set_servo_position(0, 10);
gripper(1);

camera_open_black();
camera_update();
}

```

```

int num_green = 0; //channel#0
int num_blue = 0; //channel#1

int done=0;

//repeat until done
while(done==0){
    camera_open_black();
    camera_update();

    //count color block is found
    num_green = get_object_count(0);
    num_blue = get_object_count(1);
    console_clear();

    printf("Channel 0 found %d green cube\n", num_green);
    printf("Channel 1 found %d blue cube\n", num_blue);
    printf("There is %d channel\n", get_channel_count());

    int channel_green = 0;
    int object_green = 0;

    //determine position of green block
    int pos_x_green = get_object_center_x(channel_green,
object_green);
    int pos_y_green = get_object_center_y(channel_green,
object_green);
    int cube_area_green = get_object_area(channel_green,
object_green);

    int channel_blue = 1;
    int object_blue = 0;

    //determine position of blue block
    int pos_x_blue = get_object_center_x(channel_blue, object_blue);
    int pos_y_blue = get_object_center_y(channel_blue, object_blue);
    int cube_area_blue = get_object_area(channel_blue, object_blue);

    printf("pos_x Green %d is at %d\n", object_green, pos_x_green);
    printf("pos_y Green %d is at %d\n", object_green, pos_y_green);
    printf("cube_area Green %d is at %d\n", object_green,
cube_area_green);

    printf("pos_x Blue %d is at %d\n", object_blue, pos_x_blue);
    printf("pos_y Blue %d is at %d\n", object_blue, pos_y_blue);
    printf("cube_area Blue %d is at %d\n", object_blue,
cube_area_blue);

    //if neither color block is found, keep the arm up, else lower the arm
    if((num_blue == 0 && num_green == 0) || (pos_x_green < 0 &&
pos_x_blue < 0)){
        set_servo_position(0, 10);
    }
}

```

```
        }else if(num_green >= 1 && (pos_x_green > 65 && pos_x_green < 95)
&& (pos_y_green>105 && pos_y_green<119)){
    msleep(500);
    //take the block
    take_it();
    //turn left
    motor(0, -20);
    motor(1, 20);
    msleep(1500);
    motor(0, 0);
    motor(1, 0);
    //put it down
    put_down();
    //turn back to initial position
    msleep(1000);
    motor(0, 20);
    motor(1, -20);
    msleep(1550);
    motor(0, 0);
    motor(1, 0);
    //task complete
    done=1;
}else if(num_blue >= 1 && (pos_x_blue > 65 && pos_x_blue < 95) &&
(pos_y_blue>105 && pos_y_blue<119)){
    msleep(500);
    take_it();
    //turn right
    motor(0, 20);
    motor(1, -20);
    msleep(1700);
    motor(0, 0);
    motor(1, 0);
    put_down();
    //turn back to intial position
    motor(0, -20);
    motor(1, 20);
    msleep(1700);
    motor(0, 0);
    motor(1, 0);
    //task complete
    done=1;
}
}//end while

}
```

5. My robot did behave as I plan but it was not consistence while it tries to stack the color block on another one. When it tries to stack the block, it won't stop lower the arm although it already stacked the block. I think it is because the sensor was not mounted properly, and it cannot determine the distance of the gripper to the other block.
6. If I were developing them as a work project, I would need test the gripper to make sure it must grab the block more consistence and secured.

The maximum weight of the arm that it would be able to lift up is also an aspect to be concerned. Since the color block given in the lab kit was very light weight, it won't be the same in the real life. We would want the robot to carry up heavy object so human do not have to perform those tasks.

7. If I were able to redesign my project, I would place the camera somewhere that it can locate the blocks better because my robot could not see the block if it come too closed to the blocks.

I also want my robot to have one or two extra degrees of freedom so it can move freely in its space and perform the tasks more precisely.

8. It would be great if students could visit the Nerve Center.