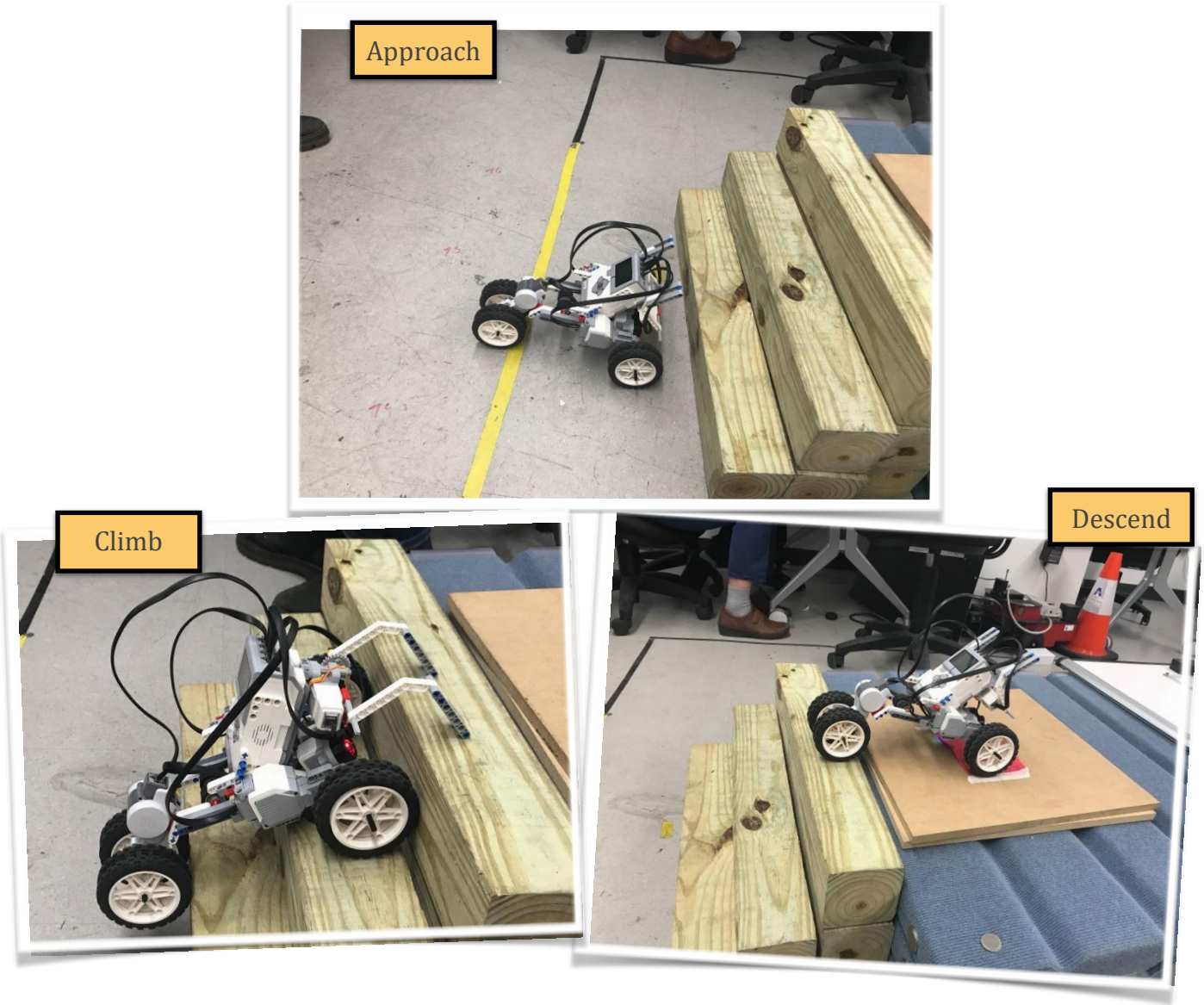
Team: Mindstorms
13 December 2017

Stair Climbing Robot

To build and program a robot that can climb up and down a set of stairs with 10 cm steps.



Setup

The environment is a set of wooden blocks arranged to form a 3-step stair. The height of each block is 10 cm. The robot is put down at a distance in front of stairs facing towards them.

The Project

Robot Design

The robot uses a light intensity sensor, and a sonar. The sonar is placed on the front side of the robot structure facing forwards for detecting the stair in front of it. It measures the distance between the robot and the nearest step. The light sensor is also placed near the sonar sensor facing downwards to detect the top of the stairs marked by a red paper. There is also a 2-hook assembly mounted at the top of the robot for pulling it up when stuck.

Behavior Strategy

Our robot starts by checking the distance from the nearest wall and adjusting the speed of the front and back wheels accordingly. When a step is detected nearby, the front wheels slow down to get a better grip on the wall. The rear wheels also speed up to push the front wheels off the ground. If the robot decides that it is stuck, it will attempt to engage the hook mechanism mounted at the top to pull itself upward to the next step. The rear wheels speed up even more when the hook is deployed to have a better chance of building momentum.

The robot keeps climbing up the stairs till the color sensor detects red color underneath after which the robot begins its descend. The strategy for descending is to gradually start moving in reverse. The tires will grip the steps avoiding a tumble.

Challenges Faced

1. Climbing up steep slope can lead to the robot getting stuck with both the wheels in the bottom corner of a different step.

2. Mounting the hook in front of robot while ensuring that the gear assembly produces enough torque to lift the robot without interfering with the rest of the climb behaviors.

3. Finding the ideal speeds at which both the front and rear wheels produce minimum slippage to get better traction.

4. Adjusting the center of gravity of the robot to ensure that it doesn’t topple over.

Final action

Once the top of the stairs is detected, the robot will descend and stop at the bottom of the staircase.

CODE

```
//============================================================================
// Name        : StairClimber.cpp
// Author      : Mindstorms
// Description : Stair Climbing Robot
//============================================================================

#include <ev3.h>
#include <string>

void climbStair() {
	OnFwdSync(OUT_AD, 11);
	OnRevReg(OUT_C, 17);
}
void decendStair() {

	while (ButtonIsUp(BTNCENTER)) {
		OnRevSync(OUT_AD, 4);
		OnFwdReg(OUT_C, 4);
		Wait(100);
	}
}

int main() {
	InitEV3();

	setAllSensorMode(US_DIST_MM, COL_COLOR, NO_SEN, NO_SEN);
	ResetRotationCount(OUT_B);
	int distance = 0, counter = 0, color;
	int prevDistance = 0;

	while (ButtonIsUp(BTNCENTER)) {
```

```
		color = readSensor(IN_2);
		distance = readSensor(IN_1);
		LcdPrintf(1, "%d ", distance);

		if (color == 5) {
			Off(OUT_ALL);
			decendStair();
		} else {
			if (counter++ > 50) {
				prevDistance = distance;
			}
			if (ButtonIsUp(BTNCENTER) && prevDistance > distance - 3 && prevDistance
< distance + 3) {
				OnRevReg(OUT_C, 19);
				RotateMotor(OUT_B, 100, 300);
				RotateMotor(OUT_B, -100, -300);
				counter = 0;
				prevDistance = 0;
			}

			if (distance > 100) {
				OnFwdSync(OUT_AD, 20);
				OnRevReg(OUT_C, 15);
			} else if (distance > 60) {
				climbStair();
			}
		}
		Wait(100);
	}

	FreeEV3();
}
```