Assignment 6

Y.S. Ginige	180195A	P.M.P.H. Somarathne	180616T
V.Y.N. Lokugama	180359G	A. Thieshanthan	180641N
W.A.V. Ravihansa	180544U	L.T.N. Wickremasinghe	180701B

Question 1

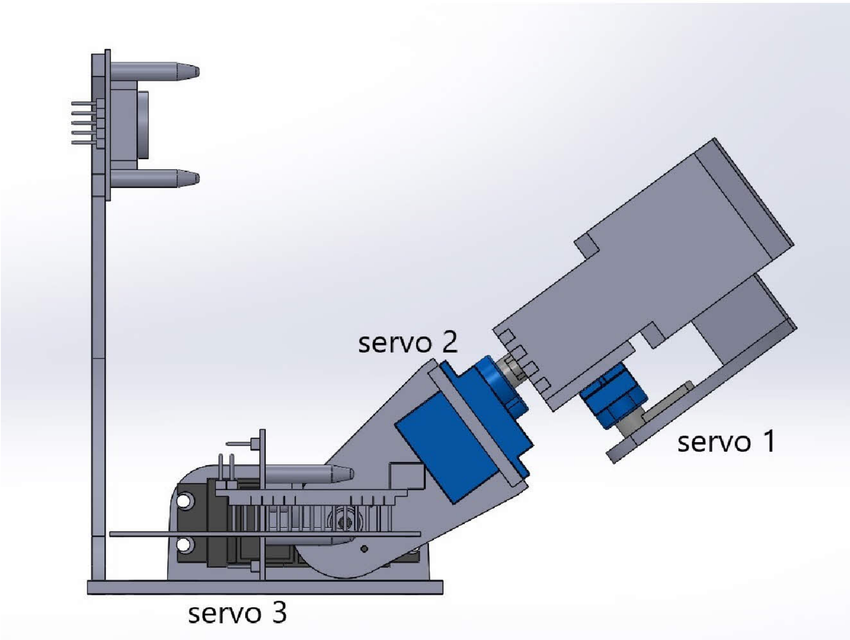

Figure 1: Side view of arm

$weight\_of\_box = 0.100kg$
$weight\_of\_servo1 = w_1$
$distance\_to\_servo1 = 5.6cm$
$weight\_of\_servo2 = w_2$
$distance\_to\_servo2 = 3.1cm$
$weight\_of\_gripping\_parts = 0.200kg$
$distance\_to\_box = 9cm$
$distance\_to\_gripper = 5.6cm$

$Torque\_required\_for\_servo3 = 0.1 \times 8 + 0.2 \times 5.6 + w_1 \times 5.6 + w_2 \times 3.1$
$Torque\_for\_servo1 = 0.05 \times 2.3 = 0.115kgcm$

With these calculations Tower pro SG90 Servo Motor can be used for servo1 and servo2. Which will set $w_1 = w_2 = 0.0147kg$. Then,

$Torque\_required\_for\_servo3 = 0.1\times8+0.2\times5.6+0.0147\times5.6+0.0147\times3.1 = 2.047kgcm$
To have 50% cushion for errors Tower Pro SG5010 can be used for servo3.

Question 2 - PID algorithm for line following

Global variables

$Error \longleftarrow 0$
$prevError \longleftarrow 0$
$deltaError \longleftarrow 0$
$sumError \longleftarrow 0$
$Kp \longleftarrow x$
$Kd \longleftarrow y$
$Ki \longleftarrow z$
$leftBaseSpeed \longleftarrow lb$
$rightBaseSpeed \longleftarrow rb$
$positiveMax \longleftarrow m$
$negativeMax \longleftarrow n$
$PIDpositiveMax \longleftarrow p$
$PIDnegativeMax \longleftarrow q$

Initializing the variables Error, prevError, deltaError, sumError to hold the values that will be updated each time getPID() function is called.

The Kp, Ki, and Kd values will be the tuning parameters of the PID control. The base speed is the speed of motors the drive the wheels of the robot when the body is in line with no error. It will be a good idea to define two base speeds to the two motors since motors may not behave identically. positiveMax and negativeMax will be the bounds we set for the speed of the motor.

PIDpositiveMax and PIDnegativeMax will be the bounds we set for the output of the getPID() function.

Basic structure of line follower code

$Repeat$
$\{$
  $pidVal \longleftarrow getPID()$
  $LeftMotorSpeed \longleftarrow validate((baseSpeed–pidVal))$
  $RightMotorSpeed \longleftarrow validate((baseSpeed + pidVal))$
  $setLeftSpeed(LeftMotorSpeed)$
  $setRightSpeed(RightMotorSpeed)$
$\}$

The getPID() will be a function that returns the change in motor speed (mapped to a scale of of PIDnegativeMax to PIDpositiveMax) that we implement in each feedback loop. We will define getPID() in a way that when that change is positive, we want a leftward turn. So, the leftmotor speed will be decreased, and the right motor speed will be increased by that change.

Assigning motor speeds will be done after validating that the base (speed+change) is within the (negativeMax, positiveMax) range. If its exceeding validate() will return the upper or lower bound.

The getPID() function

$Define\ getPID():$
$error \longleftarrow getError()$
$deltaError \longleftarrow error - prevError$
$sumError \longleftarrow update(sumError, error)$
$prevError \longleftarrow error$
$val \longleftarrow Kp * error + Kd * deltaError + Ki * sumError$
$return(map(val, (PIDnegativemax, PIDpositivemax)))$

From a closed loop feedback point of view, the sensed parameter for the controller will be an error term. Error will be positive if the robot is right of the line. Error will be negative if the robot is to the left. Getting a numerical value for the relative position of the robot will be done by getError() function.

The deltaError, sumError will correspond to the derivative and integral of the error. The update(sumError,error) function will check if the value of error is zero. If so, it should return 0. Otherwise it will return sumError+error

The value to be returned is calculated, and mapped to a range within PIDnegative-Max, and PIDpositiveMax.

The getError() function

$Define\ getError():$
$analogReadSensors()$
$return(lw_1 * L_3 + lw_2 * L_2 + lw_1 * L_1 + rw_1 * R_1 + rw_2 * R_2 + rw_3 * R_3)$

analogReadSensors will be a function to detect black or white for each sensor, then give a value 1 (black) or 0(white) for each variable L3, L2, L1, R1, R2, R3. They will correspond to the array of sensor readings in order.

The exact number of sensors may vary.
Set the coefficients $lw_1, lw_2, lw_1, rw_1, rw_2, rw_3$ so that the right sensors are weighted positively and left sensors are weighted negatively. Then when the body is to the right of the line, error will be positive, and when the body is to the left of the line, error will be negative

Question 3 - The passage of synchronized gates

After the ramp area we have to come to the first intersection line in the final task. Here we use time of flight sensor (range 3mm – 2m) to detect the gates which are synchronously close and open. Here we have three types of situations.

1. First situation → If we detect the GATE 1 is closed

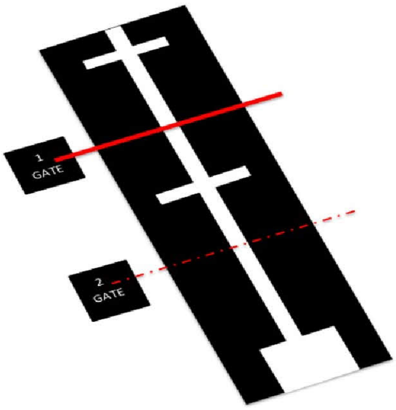

Figure 2: Situation 1

In this situation we have to wait until the GATE 1 is opened. If we detect GATE 1 is opened then after 3s the GATE 2 is also opened. Now, both gates are opened for 7s. Therefor we can go to destination with appropriate acceleration.

2. Second situation → if we detect GATE 1 is opened and GATE 2 is closed,

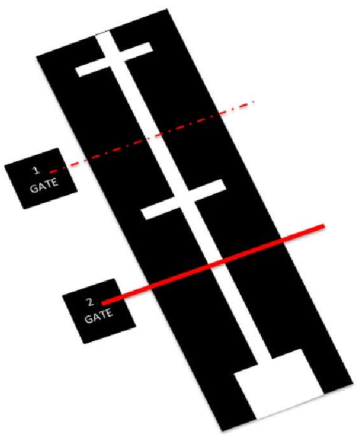

Figure 3: Situation 2

In this situation we have to wait maximum 3s. After this time (max -3s) the both GATES are opened for 7s. Therefor we can go to destination with appropriate acceleration.

3. Third situation → both GATES are opened

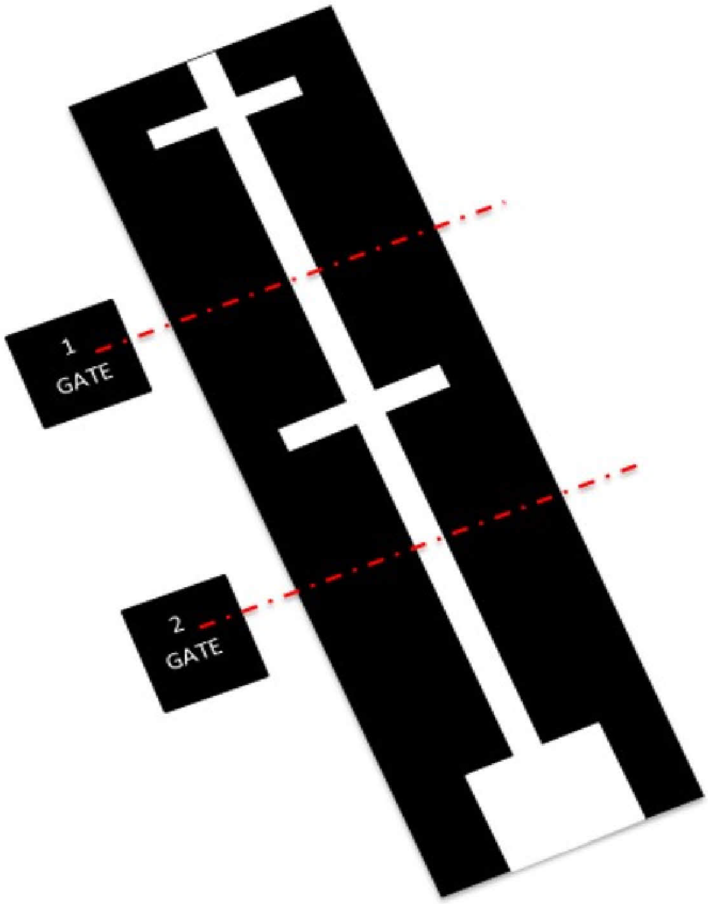

Figure 4: Situation 3

In this situation we have to wait maximum 17s for both gates are opened. After this time (max – 17s) the both GATES are opened for 7s. Therefor we can go to destination with appropriate acceleration.

In this final task we can detect the both gates from the first intersection line. Because, distance between 1st intersection line and GATE 1 is nearly 30cm and distance between 1st intersection line and GATE 1 is nearly 90cm. TOF sensor can measure these distance precisely.

According to the above three situations we complete final task when both gates are opened in same time. Therefor we have to choose appropriate acceleration for the robot.

$Minimum\ acceleration = (2 \times 120)/49 = 4.898cms^{-2}$

$Therefore,\ we\ have\ to\ maintain\ our\ robot\ acceleration \geq 4.898cms^{-2}$

	Brushed DC Motor			BLDC	
	Permanent Magnet	**Series Wound Electromagnet**	**Shunt Wound Electromagnet**	**Inrunner**	**Outrunner**
Torque	Moderate	High	Moderate	Moderate	High
Speed	Variable	Variable	Constant	High	Moderate
Phase	Single phase			Three phases	
Commutation	Using carbon brushes			Contactless electric commutation	
Rotor	Current conducting coil			Permanent magnets inside the stator	Permanent magnets outside the stator
Stator	Permanent magnets	Electromagnetic winding		Current carrying windings	
Terminals	Two - positive and negative voltages			Three - one each for each coil	
Mmagnetic Field Generation	Using permanent magnets	Using electromagnets		Using permanent magnets and electromagnets	
Angular Resolution	N/A			N/A	
Motor Complexity	Low			Moderate	
Control Mechanism	Speed controlled by the current through the motor			Energizing the windings in order. Switching frequency determines motor speed	
Control Complexity	Low - can be easily controlled by H bridge driver and PWM control for speed			High - winding energizing order and frequency must be maintained precisely	

<table>
<tr><th></th><th colspan="3">Brushed DC Motor</th><th colspan="2">BLDC</th></tr>
<tr><th></th><th>Permanent Magnet</th><th>Series Wound Electromagnet</th><th>Shunt Wound Electromagnet</th><th>Inrunner</th><th>Outrunner</th></tr>
<tr><td>Use of H-bridge</td><td colspan="3">To change the direction of rotation</td><td colspan="2">To switch between three phases</td></tr>
<tr><td>Driving Mode</td><td colspan="3">N/A</td><td colspan="2">N/A</td></tr>
<tr><td>Cost</td><td colspan="3">Low</td><td colspan="2">High</td></tr>
<tr><td rowspan="2">Advantages</td><td rowspan="2">Easy to control the speed</td><td rowspan="2">High torque</td><td rowspan="2">Self speed regulation</td><td>High speed</td><td>High torque</td></tr>
<tr><td colspan="2">No periodic maintainance</td></tr>
<tr><td rowspan="3">Disadvantages</td><td>Permanent magnets demagnetize with time</td><td rowspan="2">Poor speed regulation</td><td rowspan="2">Difficult to control the speed</td><td colspan="2" rowspan="2">Difficult to control - requires ESCs with high complexity and power</td></tr>
<tr><td>Less torque and speed over time</td></tr>
<tr><td colspan="3">Maintainance due to mechanical commutators</td><td colspan="2">Expensive</td></tr>
<tr><td>Commercialy Available Products</td><td>HANPOSE 775 80ZYT 150W 24V
CL-RS380SH</td><td>ZDY113
ECA0-series</td><td>AP231001 by ABB Motor and Mechanical Inc.</td><td>LBA2435</td><td>MTO2830-1300-S</td></tr>
</table>

	Stepper Motor			Servo Motor		
	Permanent Magnet	Variable Reluctance	Hybrid	Hobby	Winch	Industrial
Torque	High	Low	High	High	High variable	Very High
Speed	Low speeds generally used to maintain the high torque			Low constant	Moderate constant	High constant
Phase	Two phases	Many	Two phases	N/A		DC(1) or AC(1,3) phase
Commutation	Contactless electric commutation			N/A		
Rotor	Set of permenet magnets	Iron disk with teeth	Two magnetized disks with opposite poles			
Stator	Slots with windings					
Terminals	Bipolar - 4 wires Unipolar - 5/6 wires	Depends on number of phases	Bipolar - 4 wires Unipolar - 5/6 wires	Three - two for power and one for signal		Vary with manufacturer
Mmagnetic Field Generation	Using permanent magnets and electromagnets	Using electromagnets	Using permanent magnets and electromagnets	N/A		
Angular Resolution	Low	High		High		Very High
Motor Complexity	Moderate		High	N/A		High
Control Mechanism	Controller energize one or two phases at a time. Rotation is done by alligning the poles	Rotation is done by energizing the poles. Teeth of the iron disk are attracted to them.		Negative feedback loop control with a reference signal	Reference signal	Negative feedback loop control with a reference signal
Control Complexity	High			Low - internal controller circuit included		

<table>
<tr><th></th><th colspan="3">Stepper Motor</th><th colspan="3">Servo Motor</th></tr>
<tr><th></th><th>Permanent Magnet</th><th>Variable Reluctance</th><th>Hybrid</th><th>Hobby</th><th>Winch</th><th>Industrial</th></tr>
<tr><td>Use of H-bridge</td><td>To energize windings in</td><td>N/A</td><td>To energize windings in</td><td colspan="3">Not needed - internal circuit includes a H-bridge</td></tr>
<tr><td>Driving Mode</td><td colspan="3">Full step(one phase on) mode, Full step(two phase) mode, Half step mode, Micro step mode</td><td colspan="3">N/A</td></tr>
<tr><td>Cost</td><td colspan="2">High</td><td>Very high</td><td>Low</td><td>Moderate</td><td>Very High</td></tr>
<tr><td>Advantages</td><td>High torque</td><td>Very high angular resolution</td><td>High torque and angular resolution</td><td>High resolution, easy control, light-weight</td><td>Continuous rotation</td><td>High precision, high torque, high speed</td></tr>
<tr><td rowspan="3">Disadvantages</td><td rowspan="3">Low angular resolution</td><td rowspan="3">Very low torque</td><td>Larger</td><td>Limited rotation angle</td><td rowspan="3">Less angular resolution</td><td>High cost</td></tr>
<tr><td>Heavier</td><td>Less robust</td><td>Heavy</td></tr>
<tr><td>More expensive</td><td>Plastic parts wear quickly</td><td>High power dissipation</td></tr>
<tr><td>Commercialy Available Products</td><td>NEMA 34 gearbox stepper motor 24BYJ48</td><td>Tb6560 SG-PM35</td><td>28BYGE112-A-86L 57hs76 3004</td><td>SG90</td><td>Hitech HS-785HB</td><td>RMD-L-9010 (12-30V 150W)</td></tr>
</table>