Biped Patrol

Task 3.3: Think & Answer

Team Id	BP#1456
College	ADGITM
Team Leader Name	Bhaskar Dutt
e-mail	bhaskar.dragonballz@gmail.com
Date	January 15, 2020

Question No.	Max. Marks	Marks Scored
Q1	10	
Q2	20	
Q3	5	
Q4	5	
Q5	5	
Q6	10	
Q7	15	
Q8	8	
Q9	4	
Q10	8	
Q11	10	
Total	100	

Biped Patrol

Task 3.3: Think & Answer

Instructions:

- There are no negative marks.
- Unnecessary explanation will lead to less marks even if answer is correct.
- If required, draw the image in a paper with proper explanation and add the snapshot in your corresponding answer.

Q 1. Describe hardware design for the Medbot, your team is constructing. Describe various parts with well labeled image. Give reasons for selection of design. [10]

A 1. We have chosen a three layered design for building biped bot,We have decided to build a three layered bot for making our boat symmetrical and stable.
Components which we have chosen in designing of the bot layerwise is given below :

- Base layer will consist of motor cavity,tyres, Motor clamps and Two electromagnets opposite to each other.
- Middle layer will consist of Arduino Mega, Sensor-MPU6050, ZigBee module and Power distribution circuit
- Upper layer will consist of Battery.

Battery is kept on the topmost layer to maintain the weight of the whole MedBot Body.

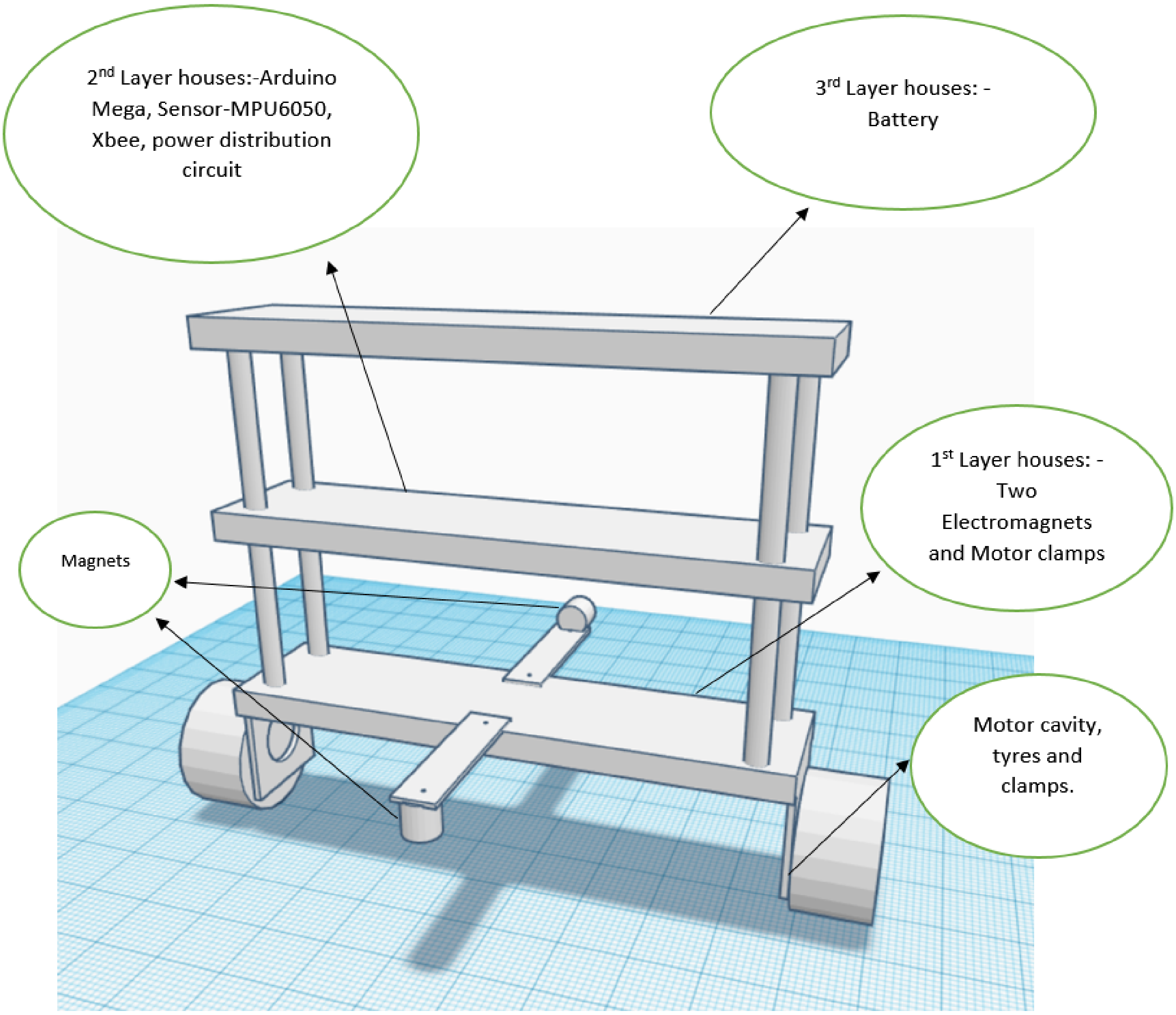

Figure 1: Labelled design

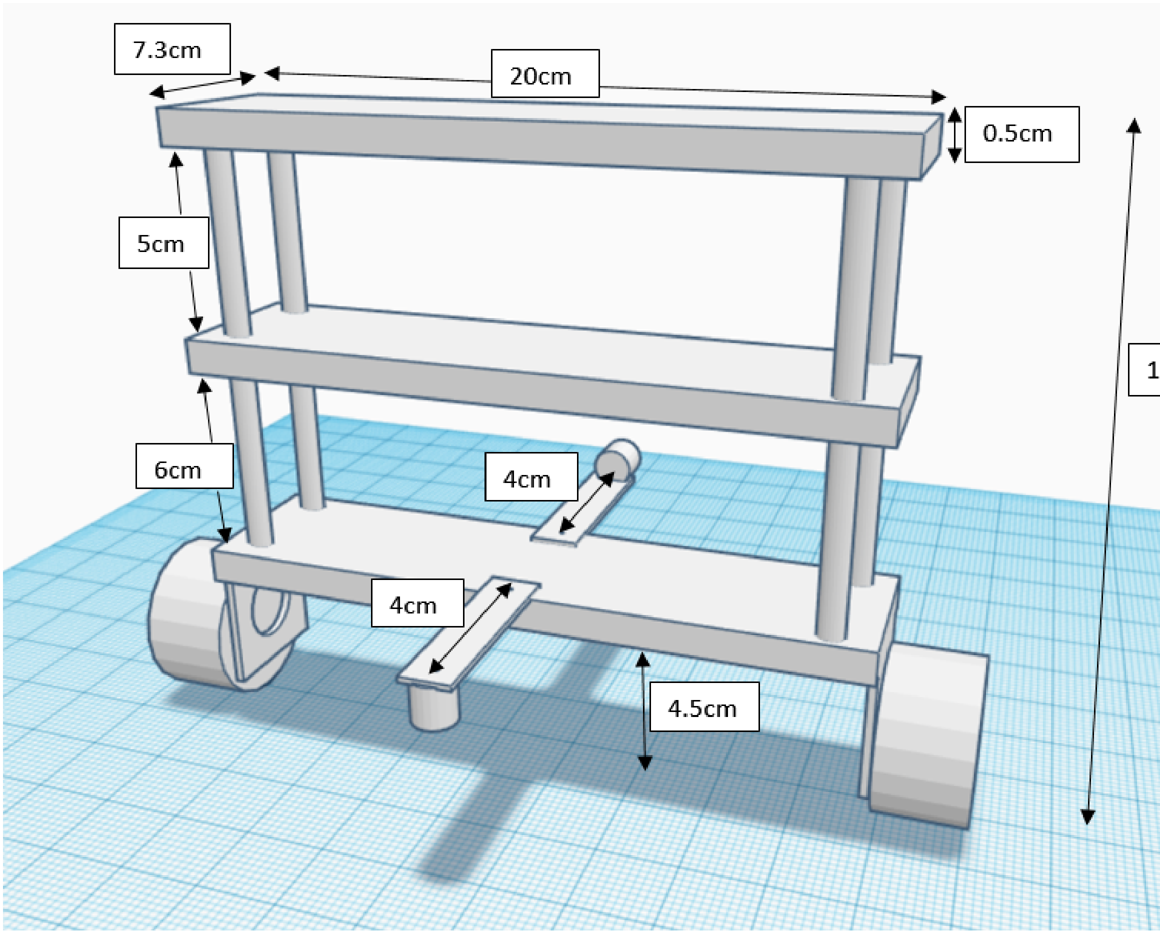

Figure 2: Bot Dimensions

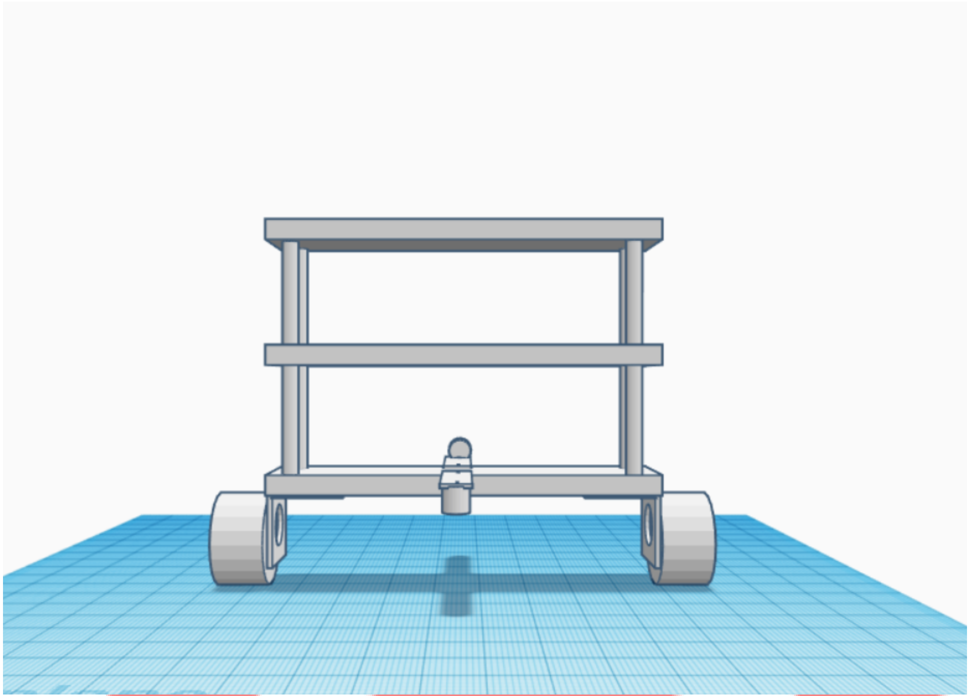

Figure 3: Front view of the bot

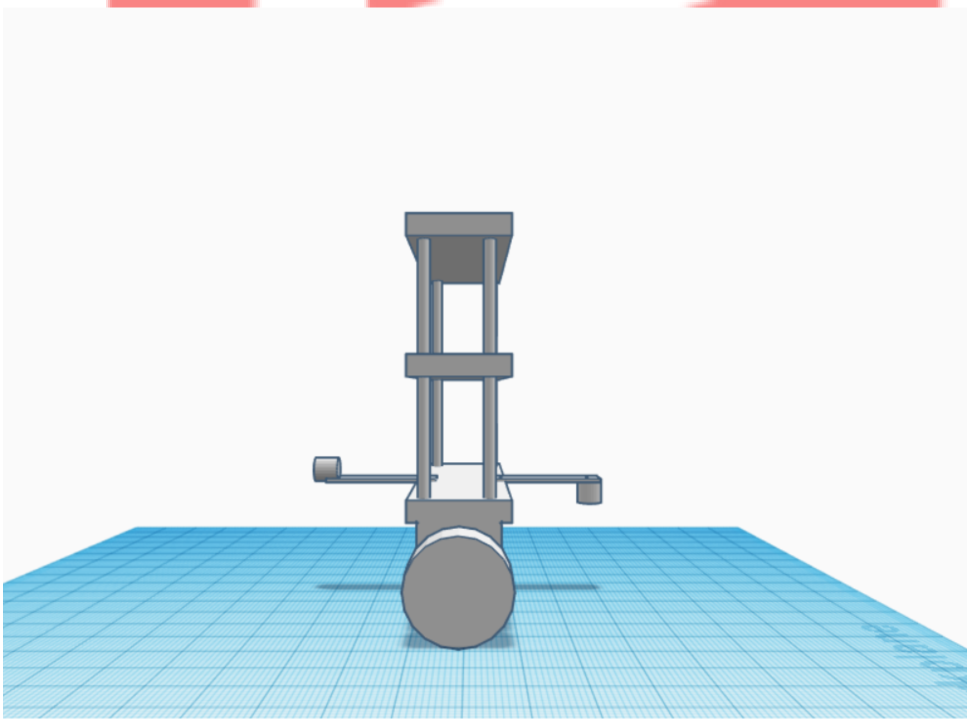

Figure 4: Left view of the bot

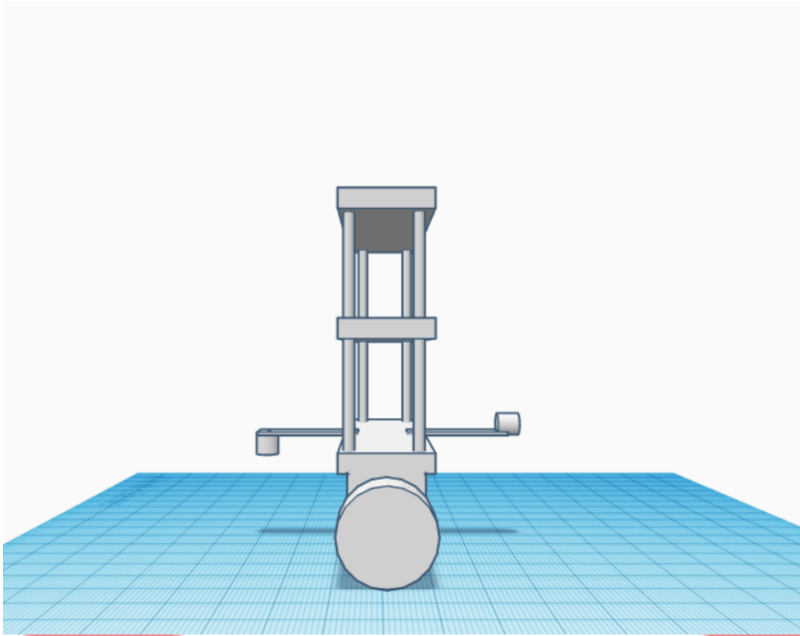

Figure 5: Right view of the bot

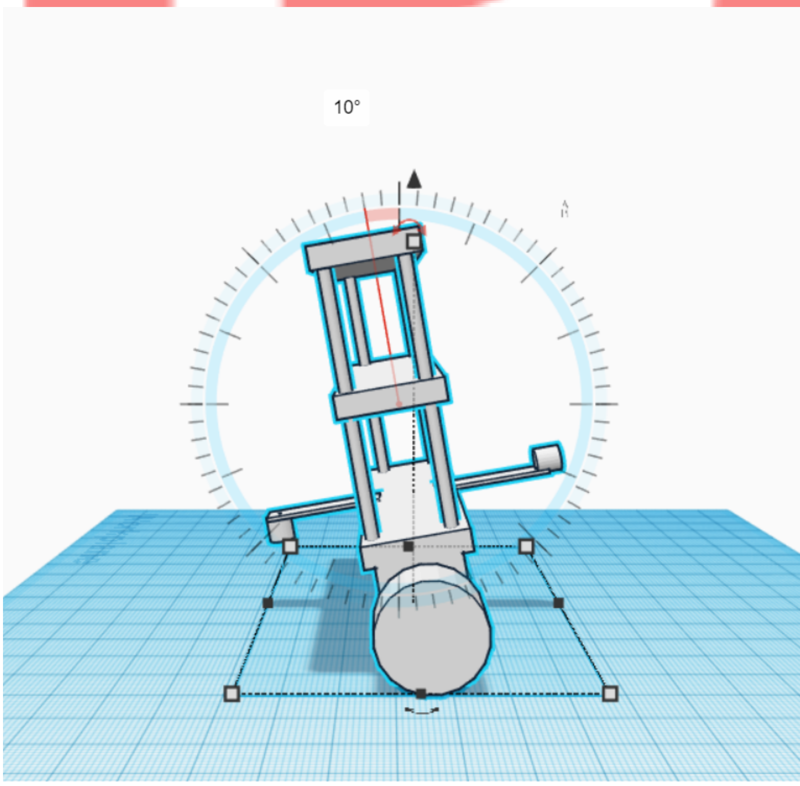

Figure 6: Right view of the bot with 10 degree tilt

Reasons for selection of this Specific multi-layered design are as follows :

- This Multi-layer selection of the MedBot will result in compact sizing of the Bot.
- MedBot will be weight balanced and symmetrical in appearance,that means center of mass will lie above a height and will be balanced.
- The ***Major advantage*** of using this design is we can lift two items at the same time which will reduce time complexity of our operation.
- MedBot is adaptable to various box sizes

Q 2. In Task 1.2, you were asked to model different systems such as Simple Pulley, Complex Pulley, Inverted Pendulum with and without input and stabilizing the unstable equilibrium point using Pole Placement and LQR control techniques. There you had to choose the states; Derive the equations (usually non-linear), find equilibrium points and then linearize around the equilibrium points. You were asked to find out the linear system represented in the form

$$\dot{X}(t) = AX(t) + BU(t) \tag{1}$$

Where $X(t)$ is a vector of all the state,i.e., $X(t) = [x_1(t), x_2(t), \ldots, x_n(t)]^T$, and $U(t)$ is the vector of input to the system, i.e. $U(t) = [u_1(t), u_2(t), \ldots, u_m(t)]^T$. A is the State Matrix & B is the Input Matrix.

In this question, you have to choose the states for the Medbot you are going to design. Model the system by finding out the equations governing the dynamics of the system using Euler-Lagrange Mechanics. Linearize the system via Jacobians around the equilibrium points representing your physical model in the form given in equation 1.
Note: You may choose symbolic representation such as M_w for Mass of wheel, etc. [20]

A 2. let
x is the position of wheels with respect to origin.
θ is the angle body makes from vertical.
$\dot{x}$ is the velocity of the wheels.
$\dot{\theta}$ is the angular velocity of the body.
ϕ is the angle turned by wheels.
m is mass of body.
m_w is mass of wheel.
m_m is mass of magnet.
$m_b = m + 2m_m$,this is mass of body, since our magnets are at equal distance from centre of mass the overall center of mass for whole body does not shift.
f is force of friction.
r is radius of wheel.

T_m is torque of the wheel.
R_a is armature resistance.
l is distance of center of mass from wheel base.
T_w is torque applied by wheel.
K_b is Kinetic energy of the body.
K_w is Kinetic energy of the wheels.
P_b is Potential energy of body.
P_w is Potential energy of wheels.
I_b is moment of inertia of body.
I_w is moment of inertia of wheels.
x_c is x co-ordinate of center of mass of the body .
y_c is y co-ordinate of center of mass of the body .

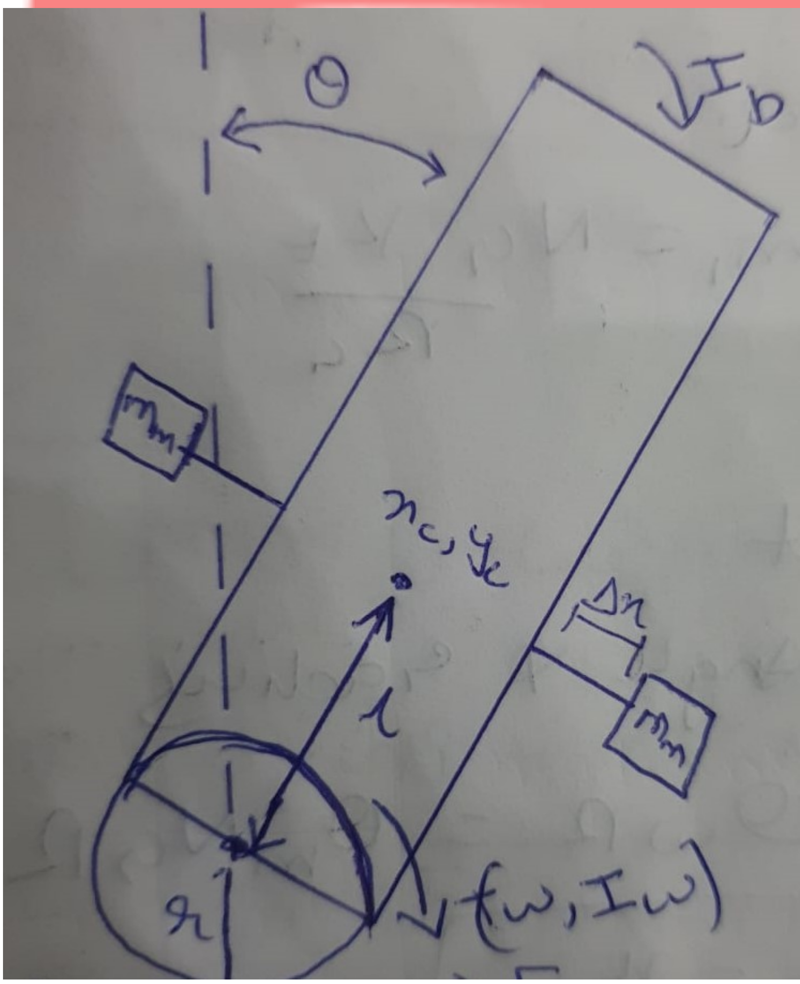

Figure 7: Mathematical model of the bot

$\dot{x} = x_1$

$\dot{x_1} = x_2$
$\dot{\theta} = \theta_1$
$\dot{\theta_1} = \theta_2$

where $x = r\phi$

$$x_c = x + lsin(\theta) \tag{2}$$

$$y_c = lcos(\theta) \tag{3}$$

now

$$K_b = \frac{I_b\dot{\theta}^2}{2} + \frac{m_b(\dot{x_c}^2 + \dot{y_c}^2)}{2} \tag{4}$$

$$K_b = \frac{I_b\dot{\theta}^2}{2} + \frac{m_b((\dot{x} + l\dot{\theta}cos(\theta))^2 + (-l\dot{\theta}sin(\theta))^2)}{2} \tag{5}$$

$$K_w = \frac{m_w\dot{x}^2}{2} + \frac{I_w\dot{\phi}^2}{2} \tag{6}$$

$$P_b = m_bglcos\theta \tag{7}$$

$$P_w = 0 \tag{8}$$

$$L = m_w\dot{x}^2 + I_w\dot{\phi}^2 + 1/2I_b\dot{\theta}^2 + 1/2m_b((\dot{x} + l\dot{\theta}cos(\theta))^2 + (-l\dot{\theta}sin(\theta))^2) - m_bglcos\theta) \tag{9}$$

Finding Euler-lagrange's equation with respect to x

$$(m_b + 2m_w + \frac{2I_w}{r^2})\dot{x_2} + (m_bl)\dot{\theta_2}cos(\theta) + m_bl\dot{\theta_1}^2sin(\theta) = \frac{2T_w}{r} \tag{10}$$

Finding Euler-lagrange's equation with respect to θ

$$(m_bl^2 + I_b)\dot{\theta_2} + m_bl\dot{x_2}cos(\theta) + m_blgsin(\theta) = -2T_w \tag{11}$$

Substituting $\dot{\theta_2}$ from eqution 11 to equation 10

$$\dot{x_2} = \frac{\frac{-(m_bl)^2gsin(\theta)cos(\theta)}{m_bl^2+I_B} + m_bl\dot{\theta_1}^2sin(\theta) + 2T_w(\frac{m_blcos\theta}{m_bl^2+I_B} + \frac{1}{r})}{2(m_w + \frac{I_w}{r^2}) + m_b + \frac{(-m_blcos(\theta))^2}{m_bl^2+I_B}} \tag{12}$$

Substituting $\dot{x_2}$ from eqution 12 to equation 11

$$\dot{\theta_2} = \frac{\frac{-(m_bl)^2(\theta_1)^2sin(\theta)cos(\theta)}{m_b+2(m_w+\frac{I_w}{r^2})} + m_bglsin(\theta) + 2(r + \frac{m_blcos(\theta)}{m_b+2(m_w\frac{I_w}{r^2})})\frac{T_w}{r}}{I_b + m_bl^2 - \frac{(m_blcos(\theta))^2}{m_b+2(m_w+\frac{I_w}{r^2})}} \tag{13}$$

Now for finding equilibrium points putting
$x_1 = 0$,$\dot{x_1} = 0$,$\theta_1 = 0$and $\dot{\theta_1} = 0$

Finding Jacobian Matrix and putting equilibrium points leaves us with the following A matrix
A= $\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & C_1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 2C_3 & 0 \end{bmatrix}$ B= $\begin{bmatrix} 0 \\ C_2 \\ 0 \\ -2C_4/r \end{bmatrix}$
where

$$C_1 = \frac{\frac{-(m_b l)^2 g\theta}{m_b l^2 + I_b}}{m_B + 2(m_w + \frac{I_w}{r^2}) - \frac{(m_b l)^2}{m_b l^2 + I_b}} \tag{14}$$

$$C_2 = \frac{\frac{1}{r} + \frac{m_b l}{m_b l^2 + I_b}}{m_B + 2(m_w + \frac{I_w}{r^2}) - \frac{(m_b l)^2}{m_b l^2 + I_b}} \tag{15}$$

$$C_3 = \frac{m_b g l}{I_b + m_b l^2 - \frac{(m_b l)^2}{m_B + 2(m_w + \frac{I_w}{r^2})}} \tag{16}$$

$$C_4 = \frac{r + \frac{m_b l}{m_b + 2(m_w \frac{I_w}{r^2}}}{I_b + m_b l^2 - \frac{(m_b l)^2}{m_B + 2(m_w + \frac{I_w}{r^2})}} \tag{17}$$

These matrices consider motor torque as an input but since we are controlling the supply given to motor we have to convert these matrices to accommodate power given to motor as input. We know that $T_m(t) = k_1 * i(t)$ and $V_a(t) = k_2 * \theta(t)$
Applying kirchoff's law in motor
$V_a(s) = Z_a * I_a + sk_b\theta$ from above equation we can write $I_a(s) = T_m/k_1$
therefore

$$V_a(s) = Z_a * T_m/k_1 + sk_b\theta \tag{18}$$

Since there is a gearbox in motor $T_w = T_m * N_g$
in real case there is some drag in the wheels due to viscosity and friction which opposes torques.
Let us consider D as drag co-efficient
Therefore wheel torque now becomes
$T_w = T_m * N_g - sD\phi$ We know that θ_m is the angle between rotor shaft and stator the more precise angle between them would be when we consider the angle of stator as well because stator is connected with the main body and it is swinging by an angleθ.
Therefore

$$T_w(s) = T_m * N_g - s(D(\theta_m - \theta))/N_g \tag{19}$$

Putting the value of T_m from equation 18 to equation 19 gives us

$$T_w(s) = \frac{N_g k_1 V_a(s)}{R_a} - s\frac{N_g k_1 k_2 \theta_m}{R_a} + s\frac{N_g k_1 k_2 \theta}{R_a} \tag{20}$$

Assume

$C_{ma} = \frac{N_g k_1}{R_a}$
$C_{mb} = \frac{N_g k_1 k_2}{R_a}$
Therefore we can re-write equation 20 as follows

$$T_w(s) = C_{ma}V_a(s) - sC_{mb}\theta_m + sC_{mb}\theta \tag{21}$$

θ_m is of no use to us and the angle of the motor shaft is related to the coordinate (x,0) of wheel axis as $x = \theta_m R_a N_g$ Hence the final equation is

$$T_w = C_{ma}V_a - \frac{sC_{mb}x_m}{rN_g} + sC_{mb}\theta \tag{22}$$

We know that torque=force*radius.Using this equation and equation 22 we can re-write Matrices A,B as follows A= $\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -C_{mb}C_2/rN_g & C_1 & C_{mb}C_2 \\ 0 & 0 & 0 & 1 \\ 0 & C_{mb}C_4/rN_g & 2C_3 & -C_{mb}C_4 \end{bmatrix}$ B= $\begin{bmatrix} 0 \\ C_{ma}C_2 \\ 0 \\ -C_{ma}C_4/r \end{bmatrix}$

$$\begin{bmatrix} x_1 \\ \dot{x_1} \\ \theta_1 \\ \dot{\theta_1} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -C_{mb}C_2/rN_g & C_1 & C_{mb}C_2 \\ 0 & 0 & 0 & 1 \\ 0 & C_{mb}C_4/rN_g & 2C_3 & -C_{mb}C_4 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ C_{ma}C_2 \\ 0 \\ -C_{ma}C_4/r \end{bmatrix} \mathrm{V}_a$$

Q 3. Equation 1 represents a continuous-time system. The equivalent discrete time system is represented as:

$$X(k+1) = A_d X(k) + B_d U(k) \tag{23}$$

Where $X(k)$ is a measure of the states at k_{th} sampling instant,i.e., $X(k) = [x_1(k), x_2(k), \ldots, x_n(k)]^T$, and $U(k)$ is the vector of input to the system at k_{th} sampling instant, i.e. $U(k) = [u_1(k), u_2(k), \ldots, u_m(k)]^T$. A_d is the Discrete State Matrix & B_d is the Discrete Input Matrix.

What should be the position of eigen values of A_d for system to be stable.
Hint: In frequency domain, continuous-time system is represented with Laplace transform and discrete-time system is represented with Z transform. [5]

A 3. From the solution of the previous question we know that,
A= $\begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -C_{mb}C_2/rN_g & C_1 & C_{mb}C_2 \\ 0 & 0 & 0 & 1 \\ 0 & C_{mb}C_4/rN_g & 2C_3 & -C_{mb}C_4 \end{bmatrix}$
and
B= $\begin{bmatrix} 0 \\ C_{ma}C_2 \\ 0 \\ -C_{ma}C_4/r \end{bmatrix}$

To obtain the Discrete from equation we will apply the Euler's Forward Rectangular Method,

$$A_d = (I + AT) \tag{24}$$

where, I= $\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
and Sampling period, T= 0.10 s
So,

$$A_d = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} + \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -C_{mb}C_2/rN_g & C_1 & C_{mb}C_2 \\ 0 & 0 & 0 & 1 \\ 0 & C_{mb}C_4/rN_g & 2C_3 & -C_{mb}C_4 \end{bmatrix} * 0.10 \tag{25}$$

$$A_d = \begin{bmatrix} 1 & 0.10 & 0 & 0 \\ 0 & 1 - 0.10(C_{mb}C_2/rN_g) & 0.10C_1 & 0.10(C_{mb}C_2) \\ 0 & 0 & 1 & 0.10 \\ 0 & 0.10(C_{mb}C_4/rN_g) & 0.20C_3 & 1 - 0.10(C_{mb}C_4) \end{bmatrix} \tag{26}$$

For a matrix to be stable its eigen values must be negative so that its response decay with time and reach a stable state.If the eigen values are positive then its response will be exponentially increasing and it will reach infinity hence making the system unstable.
We can get the eigen values of matrix A_d by solving equation$|A_d - \lambda I| = 0$.
Where
$\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 =$ sum diagonal elements of A$_d$*and*

$\lambda_1 * \lambda_2 * \lambda_3 * \lambda_4 = |A_d|$
At the moment we have$\lambda_1 = 1$ which shows that our system is unstable to make it stable we have to use some type of controller so that we can shift this unstable eigen value to the left half of complex plane.

Q 4. Will LQR control always works? If No, then why not? and if Yes, Justify your answer.
Hint: Take a look at definition of Controllable System. What is controllability? [5]

A 4. The LQR controller is simple. No specific information about the system dynamics is required. While estimating the dynamics with finite differences, a decent system model that could be sampled is required. But since the system depends on the fact that the system has to be considered linear for the sake of applying LQR, the more non-linear the system, of course, the less effective a LQR will be. Thus the lesser the controllability of the system the lesser chance of LQR being effective.

Q 5. For balancing robot on two wheel i.e. as inverted pendulum, the center of mass should be made high or low? Justify your answer. [5]

A 5. The centre of mass should be made high relative to the wheel axles because a higher centre of mass means a higher mass moment of inertia, which corresponds to a lower angular acceleration which would slow down the fall for a controlled forward movement.

$$T = Ia \tag{27}$$

T = torque, around a defined axis (Nm)
I = moment of inertia (kgm2)
$a = angularacceleration(radians/s2)$

Q 6. Why do we require filter? Do we require both the gyroscope and the accelerometer for measuring the tilt angle of the robot? Why? [10]

A 6. Filters work on the measurements that contain noises which contribute to the error in the measurement. It will then estimate the state of the system, on the basis of the current and previous states, which gives a more precise state.

For a self balancing robot the gyroscopes and accelerometers are used for measuring the tilt angle. The problem with accelerometers is the white noise during the measurement of the gravitational acceleration. Whereas the measurement from the gyroscope drifts with time – just like a spinning wheel-gyro will start to fall down when it is losing speed.

The accelerometer and gyroscope individually bring strong advantages to a navigation system; however, both have areas of data uncertainty. With both of these sensors collecting data on the same phenomena, which is the movement of an object, merging the output data to get the best of both sensors is a good option. This can be accomplished with a sensor fusion strategy. Sensor fusion techniques combine sensory data from disparate sources and generate information that has less uncertainty, or more accuracy. In the case of gyroscopes and accelerometers, they each serve to offset the other's noise and drift errors to provide accurate movement tracking. This action of combining these sensors' outputs is realized through the implementation of a filter. The filter combines information in the presence of uncertainty. In a dynamic system, a filter is ideal for systems that are continuously changing.

Q 7. What is Perpendicular and Parallel axis theorem for calculation of Moment of Inertia? Do you require this theorem for modelling the Medbot?Explain Mathematically. [15]

A 7. ***Perpendicular axis Theorem*** - The moment of inertia of a plane lamina about an axis perpendicular to its plane is equal to the sum of the moments of inertia of the lamina about any two mutually perpendicular axes, passing in its own plane, intersecting each other at the point through which the perpendicular axis passes. This theorem is restricted to plane or 2-D lamina only.

$$I_Z = I_X + I_Y$$

Where X,Y and Z are mutually perpendicular axis.
Conditions to apply this theorem –

- X, Y and Z axis should be mutually perpendicular.
- X Y axis should be in plane of body.
- Z should be perpendicular to body.

Parallel Axis Theorem – The theorem determines the moment of inertia of a rigid body about any given axis, given that moment of inertia about the parallel axis through the center of mass of an object and the perpendicular distance between the axes. It can be applied to 2-D and 3-D both.

$$I_{REQUIRED} = I_{CENTERofMASS} + MR^2$$

Where,

$$I_{REQUIRED} = Moment\,of\,inertia\,about\,any\,axis$$

$$I_{CENTERofMASS} = Moment\,of\,Inertia\,about\,axis\,passing\,through\,center\,of\,mass$$

M= Mass of the body R = Distance of the axis from center of mass
Conditions to apply this theorem –

- 2 axis must be parallel
- 1 axis should pass through center of mass
- R is perpendicular distance

Yes, during the calculation of moment of inertia of main body (I_b) the mass of various components like battery,ArduinoMega,Motor drivers etc is distributed all over the chassis and we cannot simply consider them as concentrated as it might induce errors in our model of the bot3, due to this reason we calculate the moment of inertia for every element and then apply parralel axis theorem to get net moment of inertia of the whole body around the axis of center of mass. LQR controller will help in stabilizing of MedBot from any arbitrary position to the given setpoint about axis of center of mass.

$$I_b = (I_{bp} + M_{bp}r_{bp}^2) + (I_{ar} + M_{ar}a_{r}^2) + ...(otherelements) \tag{28}$$

where bp is for battery pack and ar for arduino
Since all our elements are placed in a straight line i.e. from top to bottom their center of masses also lie in same line and we can simply use parralel axis theorem for net moment of inertia.

Q 8. What will happen in the following situations:

(a) Medbot picks a First-Aid Kit from the shelf of Medical Store but the First-Aid Kit falls inside the store. Will there be any penalty imposed, points awarded? Will the First-Aid Kit be repositioned? [2]

(b) Medbot picks a First-Aid Kit from the shelf of Medical Store but the First-Aid Kit falls outside the store. Will there be any penalty imposed, points awarded? Will the First-Aid Kit be repositioned? [2]

(c) Medbot picks a First-Aid Kit from the shelf of Medical Store but the First-Aid Kit and the Medbot both fall inside the store. Will there be any penalty imposed, points awarded? Will the First-Aid Kit be repositioned? [2]

(d) Medbot picks a First-Aid Kit from the shelf of Medical Store but the First-Aid Kit and the Medbot both fall inside the store. Will there be any penalty imposed, points awarded? Will the First-Aid Kit be repositioned? [2]

A 8. (a). No penalty will be imposed and no points will be awarded. No, the First-Aid Kit won't be repositioned.

(b). No penalty will be imposed and no points will be awarded. Yes, the First-Aid Kit won't be repositioned.

(c). No points will be awarded but a penalty will be imposed due to its fall wherein FC will be incremented by 1 and thus a fall penalty mark, MFP of 50 will be imposed. No, the First-Aid Kit won't be repositioned.

(d). No points will be awarded but a penalty will be imposed due to its fall wherein FC will be incremented by 1 and thus a fall penalty mark, MFP of 50 will be imposed. Yes, the First-Aid Kit will be repositioned.

Q 9. What will be the points awarded if Medbot picks only one of the item from the medical store and repeatedly moves back and forth around the gravel pathway or the bridge for the entire run. [4]

A 9. The points will be different for the Gravel and the Bridge path. Since in either cases the successful pickup will be counted for the pickup of one item, thus PUC will be incremented by 1 for both the cases.

$$PUC = 1 \tag{29}$$

$$MPU = 20 * PUC \tag{30}$$

Therefore,

$$MPU = 20 \tag{31}$$

If the Medbot goes back and forth in the Gravel path then,

$$LRG = 1 \tag{32}$$

since only one item has been picked

$$ERG = 3 \tag{33}$$

i.e. it's maximum value

$$MG = 50 * (0.5 * 3 + 1) \tag{34}$$

$$MG = 125 \tag{35}$$

Therefore, points scored if the Medbot moves back and forth in the Gravel path after one successful pickup = MPU + MG = 20 + 125 = 145
If the Medbot goes back and forth in the Bridge path then,

$$LRB = 1 \tag{36}$$

since only one item has been picked

$$ERB = 2 \tag{37}$$

i.e. it's maximum value

$$MB = 70 * (0.5 * 2 + 1) \tag{38}$$

$$MB = 140 \tag{39}$$

Therefore, points scored if the Medbot moves back and forth in the Bridge path after one successful pickup = MPU + MB = 20 + 140 = 160

Q 10. What are the different communication protocols you'll be using? Name the hardware interfaced related to each of the communication protocols. Explain how these communication protocols works and what are the differences between them. [8]

A 10. The different protocols which we are using in making of MedBot are-
1. **UART** (Universal Asynchronous Receiver Transmitter)
2. **Zigbee** Protocol
3. **I2C** Protocol

The hardware which are interfaced using these protocols are –
i. **UART** - This protocol is used in Arduino PC communication to transmit/ receive the data

at specific baud rate.
ii. **Zigbee Protocol** - The protocol is used in:
a). XBee – XBee communication
b). XBee - Arduino communication
iii. **I2C Protocol** - This protocol is used for Arduino – GY-87 communication.

Working of UART protocol –

a) Universal Asynchronous Receiver Transmitter is a serial communication protocol that helps us to send/receive data of Arduino to PC.

b) Each character data byte is placed between start and stop bits. Start bit is always low and stop bit is always high.

d) ATMega2560(Arduino Mega) has inbuilt USART which helps to serially communicate with PC and various serial devices. It communicates with serial devices over digital pins Tx and Rx on the board. While it communicates with PC over USB.

e) The LED attached to both digital pins determines the state of the board that either it is transmitting or receiving at one time.

f) Serial communication in Arduino over Tx/Rx pin is based on TTL logic levels.

g) Data bus sends the data to the transmitting UART in parallel form, the transmitting UART will form that data into the data packet by adding a start bit, stop bit and a parity bit. In next step the data packet is serially output bit by bit at Tx pin. The receiving UART will receive the data at its Rx pin serially bit by bit. The receiving UART than convert data into parallel form and removes the start bit, stop bit and parity bit.

Working of Zigbee Protocol
–
Zigbee is an IEEE 802.15.4 based, low power, low data rate supporting wireless networking standard. It is used as a two-way communication protocol between various sensors and control system. A short range communication from 10 to 100 meters can be performed efficiently at about 250kbps.
Architecture of ZigBee is given below -
We have used Unicast transmission in our Xbee Network. Network developed between Xbee modules is personal area networks or PANs.

Xbees have two types of address static address (64 bit address) and dynamic address (16 bit address). Devices on the same zigbee network must share the same 64 bit and 16 bit PAN IDs.

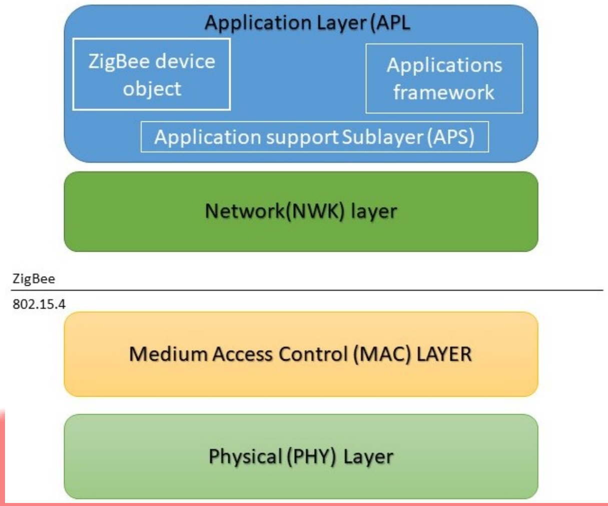

Figure 8: Architecture of ZigBee

Communication between the zigbees as a coordinator and device or peer to peer. Data transfer can be done in two ways –
a) Beacon enabled
b) Non – Beacon enabled

In Beacon enabled when the both zigbees are synchronized by sending a beacon signal from coordinator to each device, such that each device will synchronizes its clock. Once the device is synchronized it can transmit data to the coordinator using Carrier Sense Multiple Access with Collision Avoidance (CSMA-CA) method. wherein type of occupying signal is determined, or during the GTS allocation period. On sending a request, the Coordinator sends back the acknowledgement. For transfer of data from Coordinator to device, an indication is sent with the Beacon message to the device. The device then receives this indication and sends a data request message. The Coordinator sends an acknowledgment of this data request receipt and transfers the corresponding data.

In Non-Beacon Networking, the Coordinator does not transmit any Beacon message. Rather, each device transmits data using CSMA-CA method in the same frequency channel. The device transmits the data as soon as the channel is clear.

Working of I2C protocol –
I2C helps to connect multiple slaves to a single master or there are multiple masters controlling a single slave. I2C uses two wires to transmit the data. The wire connects SCL and SDA pins.

SCL Pin- Line that carries the clock signal
SDA Pin- Line to send or receive data for master and respective slaves.

a) We transfer the data in form of message. Messages are broken up into the frames of data. Message to be transmitted contain the address of the slave and one or more than one data frames that contains data to be transmitted. The message also includes start and stop conditions, read/write bits, and ACK/NACK bits between each data frame.

b) The master switches the SDA line from high to low for start condition to all connected slaves before switching the SCL line from high to low.

c) The master sends each slave the 7 or 10 bit address of the slave it wants to communicate with, along with the read/write bit.

d) Each slave compares the address sent from the master to its own address. If the address matches, the slave returns an ACK bit by pulling the SDA line low for one bit. If the address from the master does not match the slave's own address, the slave leaves the SDA line high.

e) The master sends or receives the data frame. After each data frame has been transferred, the receiving device returns another ACK bit to the sender to acknowledge successful receipt of the frame.

f) To stop the data transmission, the master sends a stop condition to the slave by switching SCL high before switching SDA high.

Figure 9: Comparision Between Zigbee, I2C and UART

PARAMETERS	**ZigBee**	**I2C**	**UART**
Nature	RF or Wi-fi based	Wire based	Wire based
No. of Wires needed	0	2	2
Data Rate	High upto 250kbps	Higher, depends upon the versions form 100kbps to 3.4Mbps	low
Type of communication	Synchronous	Synchronous	Asynchronous
Clock	No common clock is used	Common clock SCL is required.	No common clock is required.
Hardware complexity	Higher	high	low
Advantages	It is a short working period result in power saving and power consumption of communication. Supports large no. of nodes. Provides a reliable Network.	It uses open collector bus concept. Hence there is bus voltage flexibity on the interface bus. Uses flow control	It is simple communication and most popular which is available due to UART support in almost all the devices with 9 pin connectors. It is also referred as RS232 interface
Disadvantages	Its low speed and low complexity Its high maintenance cost, lack of total solution, and slow materialization.	Increases complexity of the circuit when number of slaves and masters increases. I2C interface is half duplex. Requires software stack to control the protocol and hence it needs some processing overheads on microcontroller/microprocessor.	They are suitable for communication between only two devices. It supports fixed data rate agreed upon between devices initially before communication otherwise data will be garbled.

Q 11. Why do we require IRF540N? Provide circuit diagram for interfacing IRF540N with the microcontroller. [5+5]

A 11. The IRF540N is a Logic Level MOSFET i.e we can switch it by using 5v easily available from Arduino. The MOSFET is preferred for switching is due to the following reasons :

1. MOSFETs provide greater efficiency while operating at lower voltages.
2. The absence of gate current results in high input impedance producing high switching speed.
3. They operate at lower power and draw no current.

4. They are inexpensive.

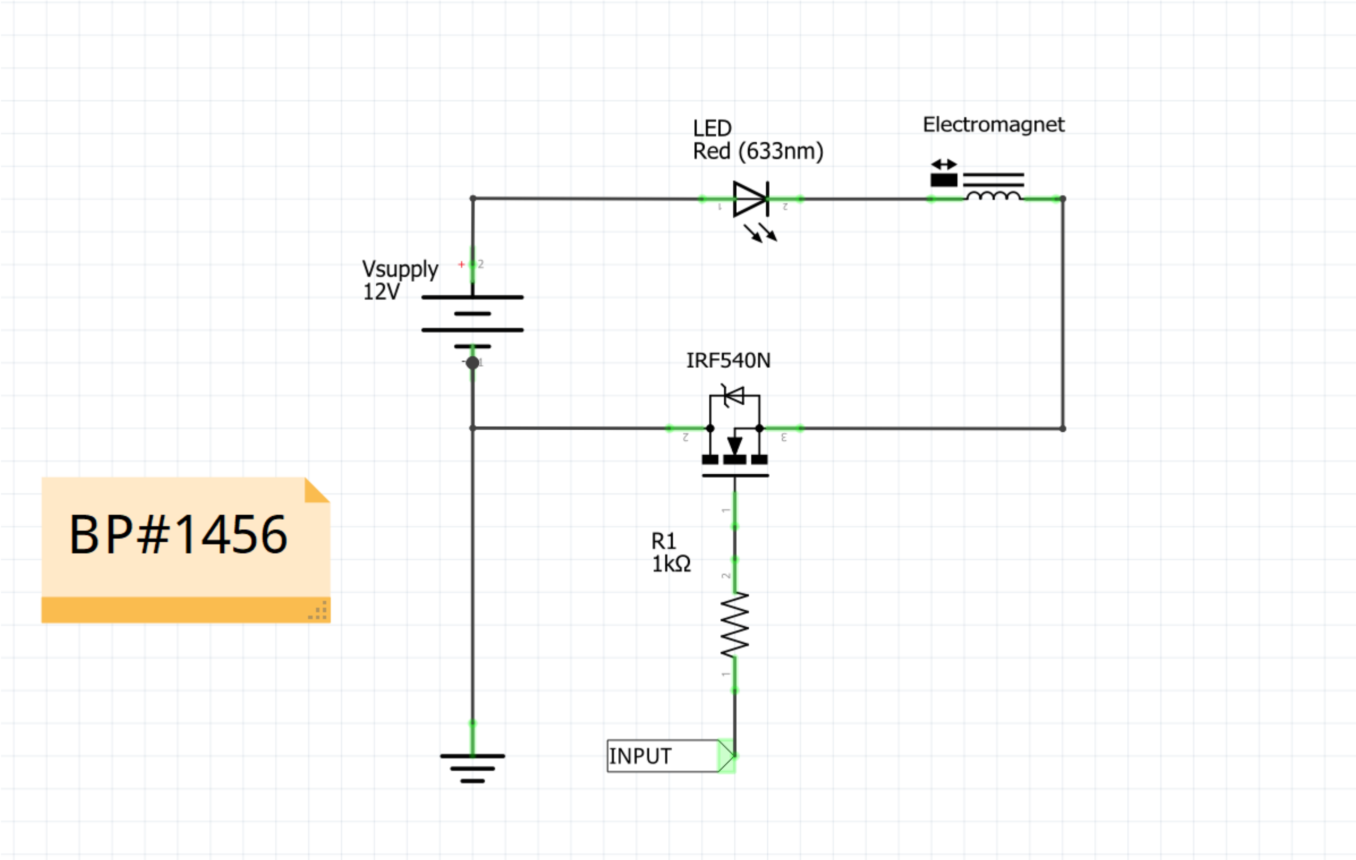

Figure 10: Circuit for switching IRF540N