Designing NXT Robot For Position Control

BY

Almoutaz Abu-arqoub **Bashar Alshalalfah**

SUPERVISED BY
Dr. Abed al-qader Al-zaro

ELECTRICAL AND COMPUTER ENGINEERING DEPARTMENT
COLLEGE OF ENGINEERING AND TECHNOLOGY
PALESTINE POLYTECHNIC UNIVERSITY

HEBRON- WEST BANK
PALESTINE

2011

Palestine Polytechnic University (PPU)

Hebron- Palestine

The Senior Project Entitled:

Designing NXT robot for position control

Prepeared By:

Almoutaz Abu-arqoub **Bashar alshalalfah**

In accordance with the recommendation of the project supervisors, and the acceptance of all examining committee members , this project has been submitted to the Department of Electrical and Computer Engineering in the College of Engineering and Technology in partial fulfillment of the requirements of Department for the degree of Bachelor of Science in Engineering.

Project Supervisor **Department Chairman**

Dedication

For our family, for our parents, for our sisters and brothers, to all whom I love, to all of our loyal teacher, to all who help us to reach this level of education, and to who has made the development of this humble project, to all of my friends and my academic younger friends, to my colleges that I have learnt from them.

Acknowledgment

By the name of Allah, most gracious, most merciful
And say: “Soon will Allah observe your work, and his messenger and the believers”

All praise and glory to Allah almighty, who made this project to accomplished. We feel honored and privileged to glorify his name in the sincerest way through this small accomplishment and ask him to accept our efforts.

The project team advances the thanks and deep appreciation to whom attribute and take care of this small project, thanks to Palestine polytechnic university especially College of Engineering and Technology and its staff, whom work to graduate the generation to come and the future builders, thanks to all of our teacher in the university, special thanks to industrial automation coordinator ”Dr. Sameer khader”, thanks to the supervisor ”Dr. abed alkader alzaro” whom they devoted their efforts to graduate us in the best way, thanks to library and its staff whom helped us to the books which we need for simple project, thanks to everyone who put his hand with us to accomplish this simple project.

Abstract

The main idea of this work to design and programming that executed given orders and tasks such as moving pin in curveting path, hand slope, catching bodies of plastic, for example if we want the robot to catch a blue ball the robot we programming him by the NXT program, after that the robot moves to the required point, then the light sensors detect weather the ball has the ordered color or not, if so the robot bring it, otherwise the robot return back to the point where he launched from.

Realizing predetermined task can be achieved by uploading the program code the NXT intelligence brick then it can applied automatically

In designing at first we explain about building the frame and the structure of the robot body, then we explain the usage of the gear box which they used to increase or decrease the velocity, we talk about the pulleys and their usage, their implementation, the five sensors sound sensor, touch sensor, Ultrasonic sensor, light sensor in addition to encoder embedded to the motor, each sensor has specified own function to do, each sensor situated in the position required to do the task in proper way, the NXT brick situated in the center of the robot to convenient the communications between the transducers and the actuators, the two DC-Servo motors are positioned in the bottom part of the robot to attach the motors rotor with the wheels for moving the robot while the third one “also DC-Servo” is situated in the upper part to do the task.

The result from this work states that the robot implenent the task precisely according to the code downloaded to the Brick that’s why the usage of DC Servo motor, but We notice that somehow the encoder attached to the shaft has little error so that some errors occurs, in addition to that the gears has backslash problem while attaching, the size of the wheels has a relationship with the speed of the robot

DESIGNING NXT ROBOT FOR POSITION CONTROL

Table of Contents

Table of Figures

Table of tables

Chapter one:

Introduction and Preface

1.1 Introduction.

1.2 Preface.

1.3 Project Objectives.

1.4 Robot Background.

1.1 Introduction

In this chapter we try to let you understand the main ideas of the whole chapters of the work, and to let you recognize the robot background and their history.

In first chapter we are talking about the objective of our robot and the planning schedule then we start a preface of robots and their some backgrounds, the needs which lead the engineers to invent the robots, history of robots…etc.

In “chapter two” we show the theoretical components and the subsystems of robots which they are commonly used in building them, such that the types of actuators used, the motors, the sensors, the power sources, the encoders, controlling the robots, the method of programming…etc

In chapter three we start our project design we talking about the frame and the body of the robot, the gears used, the sensors, the pulleys, the motors that fitting our robot and their controlling method, starting, braking, speed control, the wheels, we talking about the balance, because it is a great issue should be taken into consideration in the design.

1.2 Preface

Robotics has achieved its greatest success to date in the world of industrial manufacturing. Robot arms, or *manipulators*, comprise a 2 billion dollar industry. Bolted at its shoulder to a specific position in the assembly line, the robot arm can move with great speed and accuracy to perform repetitive tasks such as spot welding and painting (figure 1.1). In the electronics industry, manipulators place surface-mounted components with superhuman precision, making the portable telephone and laptop computer possible.

Yet, for all of their successes, these commercial robots suffer from a fundamental disadvantage: lack of mobility. A fixed manipulator has a limited range of motion that depends on where it is bolted down. In contrast, a mobile robot would be able to travel throughout the manufacturing plant, flexibly applying its talents wherever it is most effective.

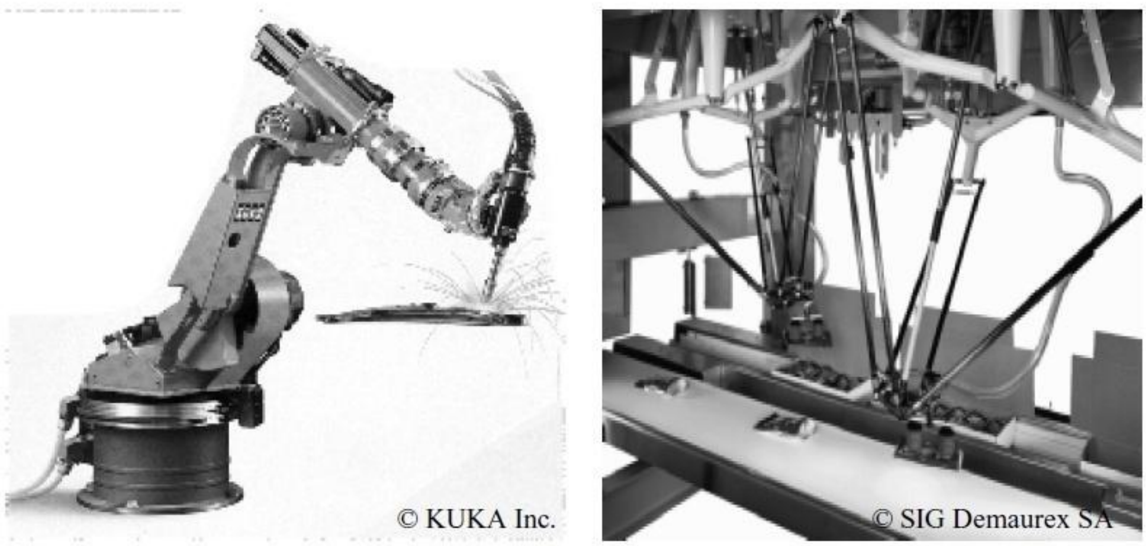


Figure 1.1 View of welding robot and parallel with robot of packaging chocolates

This project focuses on technology of mobility: how can a mobile robot move unsupervised through real-world environments to fulfill its tasks? The first challenge is locomotion itself. How should a mobile robot move and what is it about a particular locomotion mechanism that makes it superior to alternative locomotion mechanisms?

Other commercial robots operate not where humans *cannot* go but rather share space with humans in human environments. These robots are compelling not for reasons of mobility but because of their *autonomy*, and so their ability to maintain a sense of position and to navigate without human intervention is paramount.

Although mobile robots have a broad set of applications, there is one fact that is true of virtually every successful mobile robot: its design involves the integration of many different bodies of knowledge.

1.3 Project objectives

1- To achieve tasks in the factories, which facility the manufacturing process.
2- Design robot that be able to response to the request without supervising.
3- To use this robot in dangerous area to do tasks that the Human being can’t do it.

Table 1.1: Planning time for the Project Introduction

Step/weeks	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Choosing the title	/	/												
Gathering the information			/	/	/	/	/	/	/	/				
system design										/	/	/	/	
Ending the project														/

Table 1.2: Planning time for the Project.

Step/Weeks	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Purchasing the devices	/	/	/	/	/											
Building the System				/	/	/	/	/	/	/						
Test system and Modifying									/	/	/	/				
Modifying final report											/	/	/	/	/	
The Surrender																/

1.4 Robots background

1.4.1 The history of robotics

The history of robotics is one that is highlighted by a fantasy world that has provided the inspiration to convert fantasy into reality. It is a history rich with cinematic creativity, scientific ingenuity, and entrepreneurial vision. Quite surprisingly, the definition of a robot is controversial, even among robotics. At one end of the spectrum is the science fiction version of a robot, typically one of a human form-an android or humanoid-with anthropomorphic features. At the other end of the spectrum is the repetitive, efficient robot of industrial automation. In ISO 8373, the

International Organization for Standardization defines a robot as “an automatically controlled, reprogrammable, multipurpose manipulator with three or more axes.” The Robot Institute of America designates a robot as “a reprogrammable, multifunctional manipulator designed to move material, parts, tools, or specialized devices through various programmed motions for the performance of a variety of tasks.” A more inspiring definition is offered by Merriam Webster, stating that a robot is “a machine that looks like a human being and performs various complex acts (as walking or talking) of a human being.”

1.4.2 Inventions Leading to Robotics

The field of robotics has evolved over several millennia, without reference to the word *robot* until the early 20th Century. In 270 B.C., ancient Greek physicist and inventor Ctesibus of Alexandria created a water clock, called the clepsydra, or “water-thief,” as it translates. Powered by rising water, the clepsydra employed a cord attached to a float and stretched across a pulley to track time. Apparently, the contraption entertained many who watched it passing away the time, or stealing their time, thus earning its namesake. Born in Lyon, France, Joseph Jacquard (1752–1834) inherited his father’s small weaving business but eventually went bankrupt. Following this failure, he worked to restore a loom and in the process developed a strong interest in mechanizing the manufacture of silk. After a hiatus in which he served for the Republicans in the French Revolution, Jacquard returned to his experimentation and in 1801 invented a loom that used a series of punched cards to control the repetition of patterns used to weave cloths and carpets. Jacquard’s card system was later adapted by Charles Babbage in early 19th Century Britain to create an automatic calculator, the principles of which later led to the development of computers and computer programming. The inventor of the automatic rifle, Christopher Miner Spencer (1833–1922) of Manchester, Connecticut, is also credited with giving birth to the screw machine industry. In 1873, Spencer was granted a patent for the lathe that he developed, which included a camshaft and a self-advancing turret. Spencer’s turret lathe took the manufacture of screws to a higher level of sophistication by automating the process. In 1892, Seward Babbitt introduced a motorized crane that used a mechanical gripper to remove ingots from a furnace, 70 years prior to General Motors’ first industrial robot used for a similar purpose. In the 1890s Nikola Tesla—

known for his discoveries in AC electric power, the radio, induction motors, and more-invented the first remote-controlled vehicle, a radio-controlled boat. Tesla was issued Patent #613.809 on November 8, 1898, for this discovery.

1.4.3 The Birth of the Industrial Robot

Following World War II, America experienced a strong industrial push, reinvigorating the economy. Rapid advancement in technology drove this industrial wave—servos, digital logic, solid state electronics, etc. The merger of this technology and the world of science fiction came in the form of the vision of Joseph Engelberger, the ingenuity of George Devol, and their chance meeting in 1956. Joseph F. Engelberger was born on July 26, 1925, in New York City. Growing up, Engelberger developed a fascination for science fiction, especially that written by Isaac Asimov. Of particular interest in the science fiction world was the robot, which led him to pursue physics at Columbia University, where he earned both his bachelor’s and master’s degrees. Engelberger served in the U.S. Navy and later worked as a nuclear physicist in the aerospace industry.

Table 1.3: Project Balance Sheet.

Devices	**Price**
NXT Microcontroller	350$
3-Servo motors,10V	300$
5-Sensors	250$
Building frame bodies	100$
Table design and mat	350$
Rechargeable battery,6 AA	60$
Computer usage	20$
Over heads	100$
Total	1530$

Chapter Two:

Theoretical Component and Subsystem

Theoretical component and subsystem

Actuators are one of the key components contained in a robotic system. A robot has many degrees of freedom, each of which is a servoed joint generating desired motion. We begin with basic actuator characteristics and drive amplifiers to understand behavior of servoed joints. Most of today's robotic systems are powered by electric servomotors. Therefore, we focus on electromechanical actuators, beside to the sensors and encoders used in addition to the robot source and motors control by power electronics, the possible languages used in programming.

2.1 DC servo motor

Figure 2.1.1 illustrates the construction of a DC servomotor, consisting of a stator,a rotor, and a commutation mechanism. The stator consists of permanent magnets, creating a magnetic field in the air gap between the rotor and the stator. The rotor has several windings arranged symmetrically around the motor shaft. An electric current applied to the motor is delivered to individual windings through the brush-commutation mechanism, as shown in the figure. As the rotor rotates the polarity of the current flowing to the individual windings is altered. This allows the rotor to rotate continually.

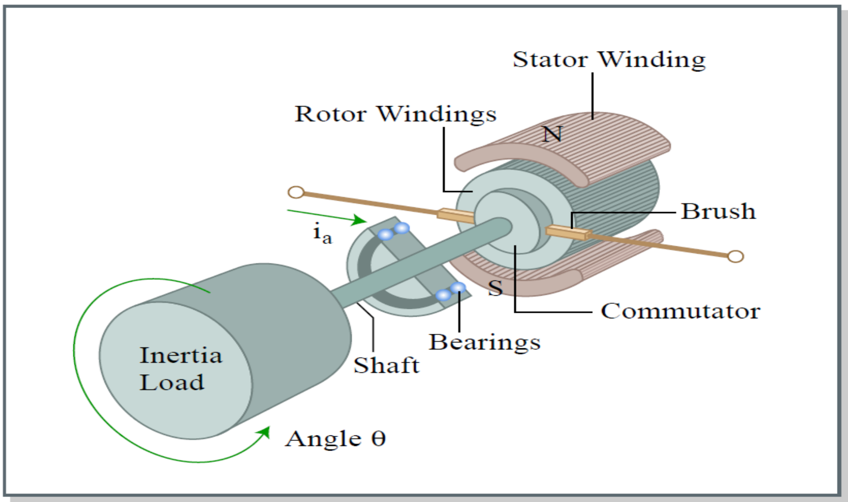


Figure 2.1.1 Construction of DC motor

Let τ_m be the torque created at the air gap, and i the current flowing to the rotor windings. The torque is in general proportional to the current, and is given by

$$\tau_m = K_t \cdot i \qquad (2.1)$$

Where the proportionality constant K_t , is called the ***torque constant,*** one of the key parameters describing the characteristics of a DC motor. The torque constant is determined by the strength of the magnetic field, the number of turns of the windings, the effective area of the air gap.

The radius of the rotor, and other parameters associated with materials properties. In an attempt to derive other characteristics of a DC motor, let us first consider an idealized energy transducer having no power loss in converting electric power into mechanical power. Let E be the voltage applied to the idealized transducer. The electric power is then given by E , i, which must be equivalent to mechanical power:

$$P_{in} = E.i = \tau_m.\omega_m \qquad (2.2)$$

Where ω_m is the angular velocity of the motor rotor. Substituting eq.(1) into eq.(2) and dividing both sides by i yield the second fundamental relationship of a DC motor:

$$E = K_{t..}\omega_m \qquad (2.3)$$

The above expression dictates that the voltage across the idealized power transducer is proportional to the angular velocity and that the proportionality constant is the same as the torque constant given by eq.(1).This voltage E is called the back emf (electro-motive force) generated at the air gap, and the proportionality constant is often called the back emf constant. Note that based on eq. (1), the unit of the torque constant is Nm/A in the metric system, whereas the one of the back emf constant is V/rad/s based on eq. (2). The actual DC motor is not a loss-less transducer, having resistance at the rotor windings and the commutation mechanism. Furthermore, windings may exhibit some inductance, which stores energy. Figure 2.2 shows the schematic of the electric circuit, including the windings resistance R and inductance L. From the figure.

$$u=R.i+L.(di/dt) + E$$

(2.4)

where u is the voltage applied to the armature of the motor.

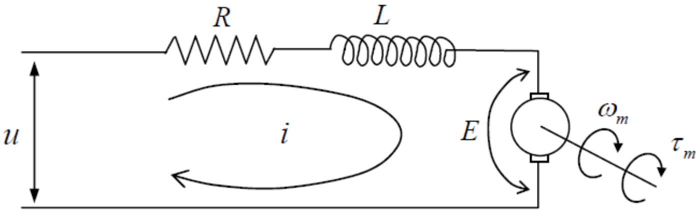


Figure 2.2: electric circuit of armature

Combining eq.(1),(3) and (4), we can obtain the actual relationship among the applied voltage u, the rotor angular velocity ω_m , , and the motor torque τ_m.

$$(K_t/R).u = \tau_m + T_e.(d\tau_m/dt) + (K_t^2/R).\omega_m \tag{2.5}$$

Where time constant $T_e=L/R$ called the motor reactance is often negligibly small. Neglecting this second term the above equation reduces to an algebraic relationship:

$$\tau_m = (K_t/R).u - (K_t^2/R).\omega_m \tag{2.6}$$

This is called the torque-speed characteristic.

That the motor torque increases in proportion to the applied voltage, but the net torque reduces as the angular velocity increases. Figure 2.3 illustrates the torque-speed characteristics The negative slope of the straight lines, $-K_t^2/R$, implies that the voltage-controlled DC motor has an inherent damping in its mechanical behavior. The power dissipated in the DC motor is given by

$$P_{dis} = R.i^2 = (R/K_t^2).\ \tau_m^{\ 2} \tag{2.7}$$

from eq.(1). Taking the square root of both sides yields

$$P_{dis}^{0.5} = \tau_m / K_m\ ,\ K_m = K_t/R^{0.5} \tag{2.8}$$

Where the parameter K_m is called the motor constant, the motor constant represents how effectively electric power is converted to torque. The larger the motor constant becomes, the larger the output torque is generated with less power dissipation A DC motor with more powerful magnets, thicker winding wires, and a larger rotor diameter has a larger motor constant A motor with a larger motor constant, however, has a larger damping, as the negative slope of the torque-speed characteristics becomes steeper, as illustrated in Figure 2.3.

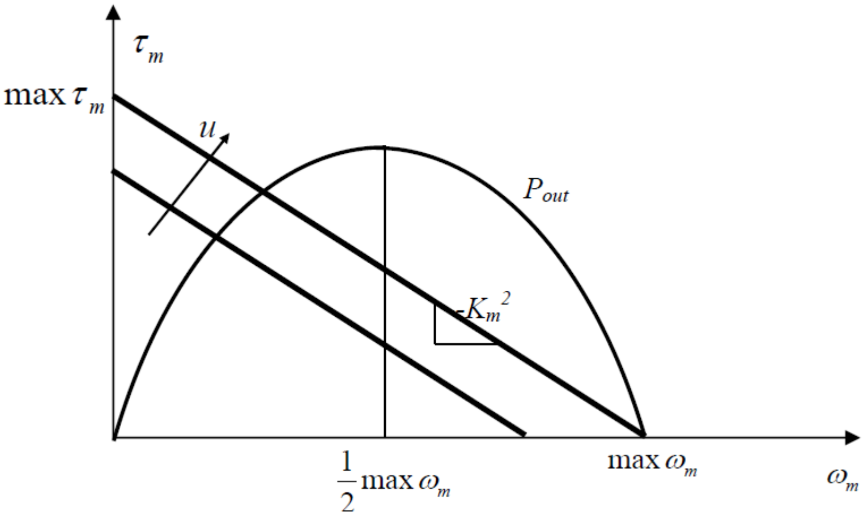


Figure 2.3: Torque-speed characteristics and output power

Taking into account the internal power dissipation, the net output power of the DC motor is given by:

$P_{out}= \tau_m.\omega_m = ((K_t/R).u-K_m^2.\omega_m).\ \omega_m$. (2.9)

This net output power is a parabolic function of the angular velocity, as illustrated in Figure 2.3. It should be noted that the net output power becomes maximum in the middle point of the velocity axis, i.e. 50 % of the maximum angular velocity for a given armature voltage *u*. This implies that the motor is operated most effectively at 50 % of the maximum speed. As the speed departs from this middle point, the net output power decreases, and it vanishes at the zero speed as well as at the maximum speed. Therefore, it is important to select the motor and gearing combination so that the maximum of power transfer be achieved.

2.1.1 Ratings and specifications

Several characteristics are important in selecting a DC motor. The first two are its input ratings that specify the electrical requirements of the motor.

Operating voltage: If batteries are the source of power for the motor, low operating voltages are desirable because fewer cells would be needed to obtain the specified voltage. However, the electronics to drive motors are typically more efficient at higher voltages.

Typical DC motors may operate on as few as 1.5 volts on up to 100 volts. Robotics often uses motors that operate on 6, 12, or 24 volts.

Operating current:

Ideally one would like a motor that produces a great deal of power while requiring a minimum of current. Typically however the current rating (in conjunction with the voltage rating) is a good indication of the power output capacity of a motor. Motors that draw more current will deliver more power. Also, a given motor draws more current as it delivers more output torque. Thus current ratings are often given when the motor is stalled. At this point it is drawing the maximal amount of current.

The next three ratings describe the motor's output characteristics:

Speed:

Usually this is specified as the speed in rotations per minute (RPM) of the motor when it is unloaded, or running freely, at its specified operating voltage. Typical DC motors run at speeds from several thousand to ten thousand RPM.

Torque:

The torque of a motor is the rotary force produced on its output shaft. When a motor is stalled it is producing the maximum amount of torque that it can produce. Hence the torque rating is usually taken when the motor has stalled and is called the stall torque. The motor torque is measured in ounce-inches (in the English system). A rating of one ounce-inch means that the motor is exerting a tangential force of one ounce at a radius of one inch from the center of its shaft. Torque ratings may vary from less than one ounce-inch to several dozen ounce- inches for large motors.

Power:

The power of a motor is the product of its speed and torque. The power output is greatest somewhere between the unloaded speed (maximum speed, no torque) and the stalled state (maximum torque, no speed).

2.1.2 Speed, Torque, and Gear Reduction

It was mentioned earlier that the power delivered by a motor is the product of its speed and the torque at which the speed is applied. If one measures this power over the full range of operating speeds”from unloaded full throttle to stall” one gets a bell-shaped curve of motor power output.

When unloaded, the motor is running at full speed, but at zero torque, thus producing zero power. Conversely, when stalled, the motor is producing its maximum torque output, but at zero speed “also producing zero power” Hence the maximum power output must lie somewhere in between.

A typical DC motor operates at speeds that are far too high to be useful, and torques that are far too low. Gear reduction is the standard method by which a motor is made useful.

Using gear reduction, the motor shaft is fitted with a gear of small radius that meshes with a gear of large radius. The motor's gear must revolve several times into order to cause the large gear to revolve once. It is evident that the speed of rotation is decreased, but, overall power is preserved (excepting losses due to friction) and therefore the torque must increase.

By ganging together several stages of this gear reduction, an immensely strong torque can be produced at the final stage.

The challenge when designing a high-performance gear reduction for a competitive robot is to determine the amount of reduction that will allow the motor to operate at highest efficiency. If the normal operating point of a motor/gear train assembly is faster than the peak efficiency point, the gear train will be able to accelerate quickly, but will not be operating at peak efficiency once it has reached the maximum velocity. Depending on the mass of the robot and the performance desired, different gear ratios might be appropriate. Experimentation is probably the best way to choose the best gear train.

2.2 Dynamics of Single-Axis Drive Systems

DC motors and other types of actuators are used to drive individual axes of a robotic system. Figure 2.4 shows a schematic diagram of a single-axis drive system consisting of a DC motor, a gear head, and arm links, An electric motor, such as a DC motor, produces a relatively small torque and rotates at a high speed, whereas a robotic joint axis in general rotates slowly, and needs a high torque to bear the load. In other words, the impedance of the actuator:

Z_m=Torque/angular velocity = τ_m/ω_m. (2.10)

is much smaller than that of the load.

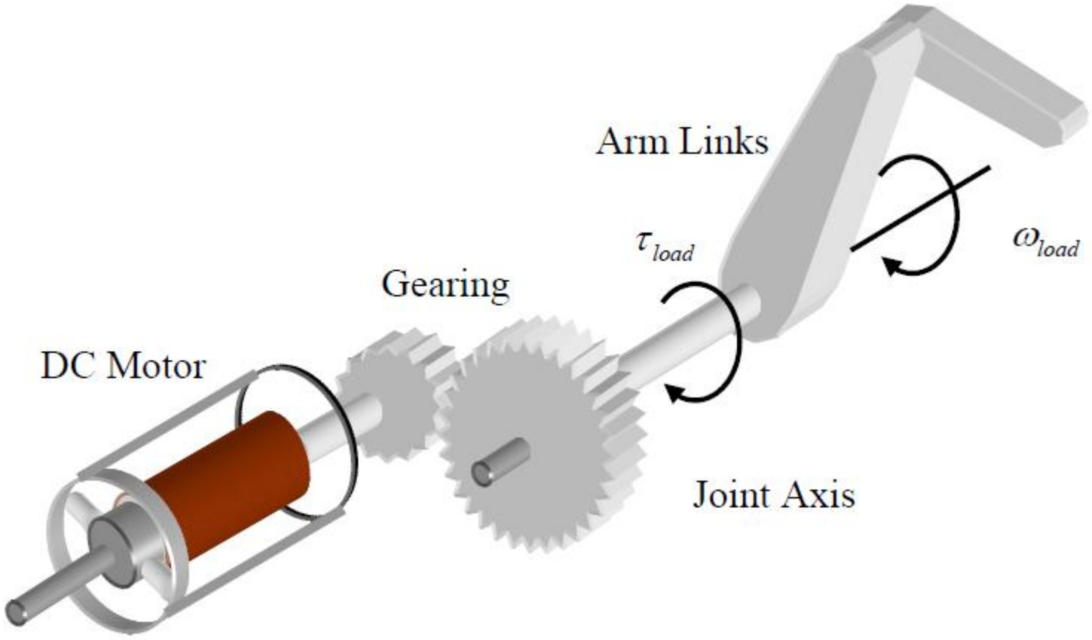


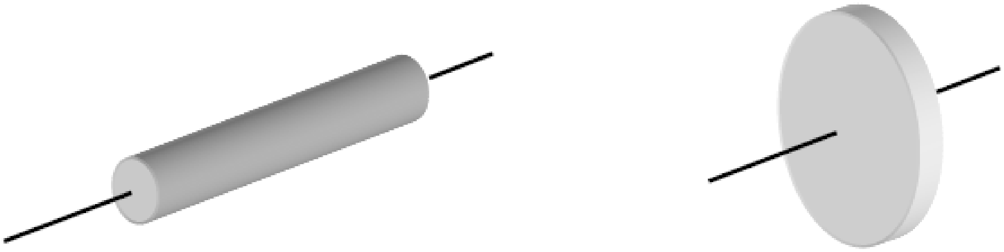
Figure 2.4: joint axis drive system

To fill the gap we need a gear reducer, as shown in Figure 2.4. Let (r > 1) be a gear reduction ratio (If *d*1 and *d*2 are diameters of the two gears, the gear reduction ratio is r = d_2/d_1). The torque and angular velocity are changed to:

$T_{load} = r.\tau_m$, $\omega_{load} = (1/r).\ \omega_m$ (2.11)

Note that the effective inertia of the motor rotor is r^2 times larger than the original value I_m when reflected to the joint axis. Likewise, the motor constant becomes r^2 times larger when reflected to the joint axis. The gear ratio of a robotic system is typically 20 ~ 100, which means that the effective inertia and damping becomes 400 ~ 10,000 times larger than those of the motor itself.

For fast dynamic response, the inertia of the motor must be small. This is crucial requirement as the gear ratio gets larger, like robotics applications. There are two ways of reducing the rotor inertia in motor design. One is to reduce the diameter and make the rotor longer, as shown in Figure 2.5-(a). The other is to make the motor rotor very thin, as a pancake, as shown in Figure(2.5)-b



(a) Long and slender (b) Pancake

Figure 2.5: two ways to reducing motor rotor inertia

Most robots use the long and slender motors as Figure (a), and some heavy-duty robots use the pancake type motor. Figure 2.2.3 shows a pancake motor by Mavilor Motors.

2.3 Power Electronic Devices

Performance of servomotors used for robotics applications highly depends on electric power amplifiers and control electronics, broadly termed power electronics. Power electronics has shown rapid progress in the last two decades, as semiconductors became faster, more powerful, and more efficient. In this section we will briefly summarize power electronics relevant to robotic system development.

2.3.1 Pulse width modulation (PWM)

In many robotics applications, actuators must be controlled precisely so that desired motions of arms and legs may be attained. This requires a power amplifier to drive a desired level of voltage (or current indirectly) to the motor armature, as discussed in the previous section. Use of a linear amplifier (like an operational amplifier), however, is power-inefficient and impractical, since it entails a large amount of power loss. Consider a simple circuit consisting of a single transistor for controlling the armature voltage, as shown in Figure 2.6. Let V be the supply voltage connected to one end of the motor armature. The other end of the armature is connected to the collector of the transistor. As the base voltage varies the emitter-collector voltage varies, and thereby the voltage drop across the motor armature, denoted u in the figure, varies accordingly. Let i be the collector current flowing through the transistor. Then the power loss that is dissipated at the transistor is given by:

$$P_{loss} = (V- u).i = 1/R.(V- u).u \tag{2.3.1}$$

Where R is the armature resistance. Figure 2.3.2 plots the internal power loss at the transistor against the armature voltage. The power loss becomes the largest in the middle, where half the supply voltage V/2 acts on the armature. This large heat loss is not only wasteful but also harmful, burning the transistor in the worst case scenario. Therefore, this type of linear power amplifier is seldom used except for driving very small motors.

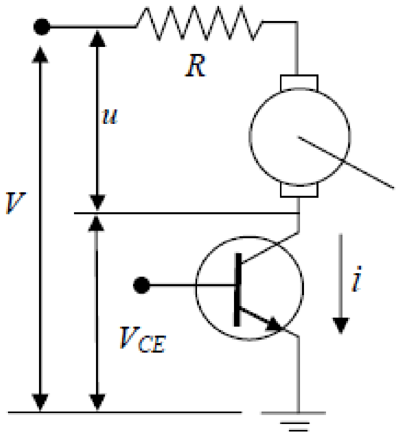


Figure 2.6: analogue power amplifier For driving the armature voltage

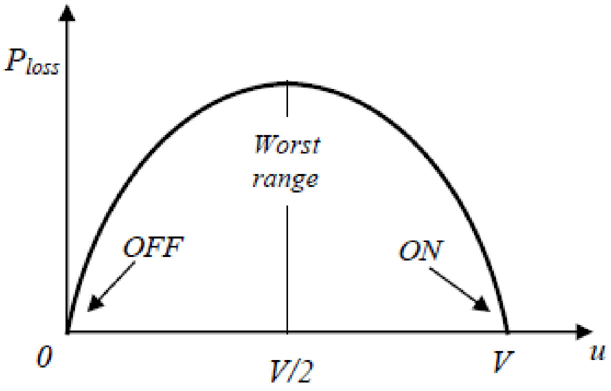


Figure 2.7: power loss at the transistor vs. the armature voltage.

An alternative is to control the voltage via ON-OFF switching. Pulse Width Modulation, or PWM for short, is the most commonly used method for varying the average voltage to the motor. In Figure 2.7 it is clear that the heat loss is zero when the armature voltage is either 0 or V. This means that the transistor is completely shutting down the current (OFF) or completely admitting the current (ON). For all armature voltages other than these complete ON-OFF states, some fraction of power is dissipated in the transistor. Pulse Width Modulation (PWM) is a technique to control an effective armature voltage by using the ON-OFF switching alone. It varies the ratio of time length of the complete ON state to the complete OFF state. Figure 2.8 illustrates PWM signals. A single cycle of ON and OFF states is called the PWM period, whereas the percentage of the ON state in a single period is called duty rate. The first PWM signal is of 60% duty, and the second one is 25 %. If the supply voltage is *V=10 volts*, the average voltage is 6 volts and 2.5 volts, respectively.

The PWM period is set to be much shorter than the time constant associated with the mechanical motion. The PWM frequency, that is the reciprocal to the PWM period, is usually 2 ~ 20 kHz, whereas the bandwidth of a motion control system is at most 100 Hz. Therefore, the discrete switching does not influence the mechanical motion in most cases.

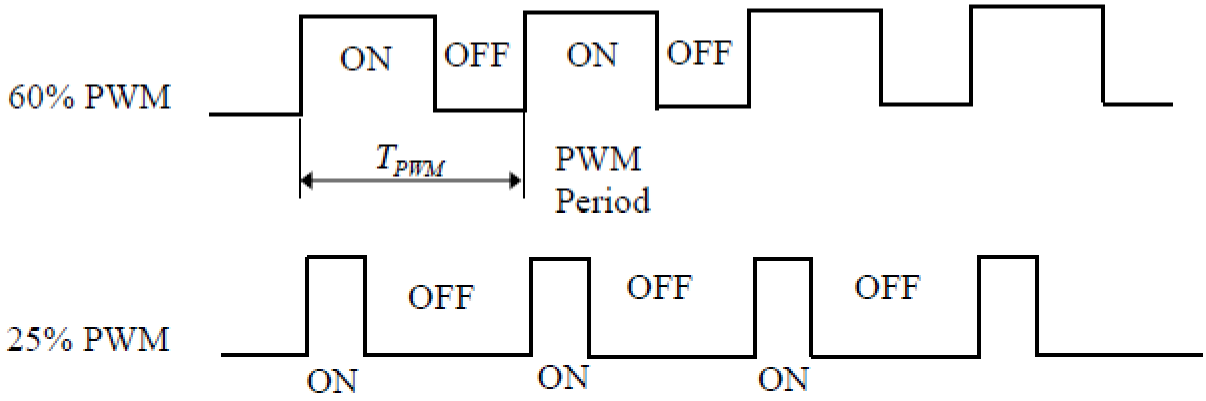


Figure 2.8: pulse width modulation

As modeled in eq.(2.4), the actual rotor windings have some inductance *L*. If the electric time constant *Te* is much larger than the PWM period, the actual current flowing to the motor armature is a smooth curve, as illustrated in Figure 2.9-(a). In other words, the inductance works as a low-pass filter, filtering out the sharp ON-OFF profile of the input voltage. In contrast, if the electric time constant is too small, compared to the PWM period, the current profile becomes zigzag, following the rectangular voltage profile, as shown in Figure 2.9-(b). As a result, unwanted high frequency vibrations are generated at the motor rotor. This happens for some types of pancake motors with low inductance and low rotor inertia.

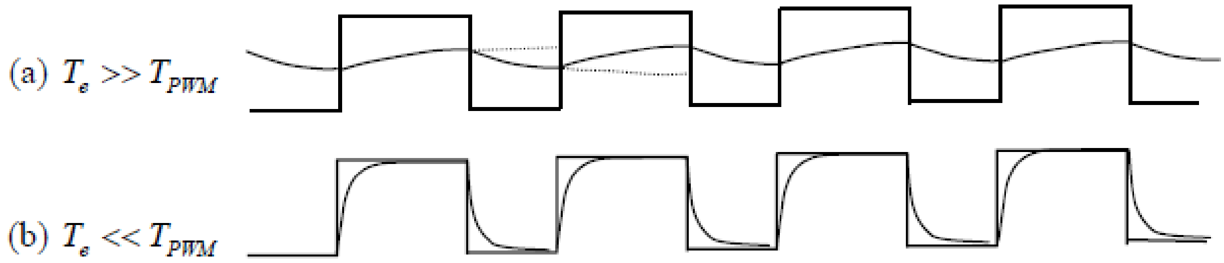
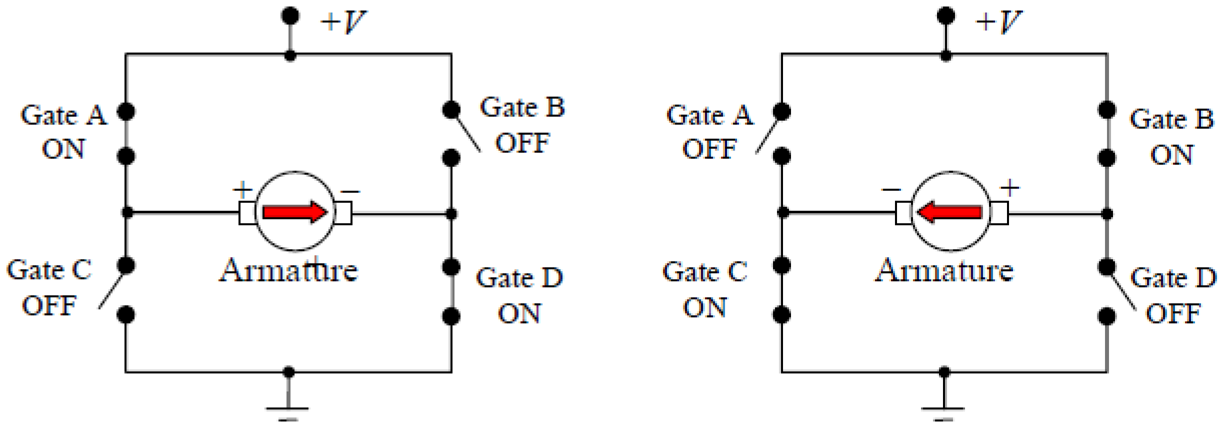


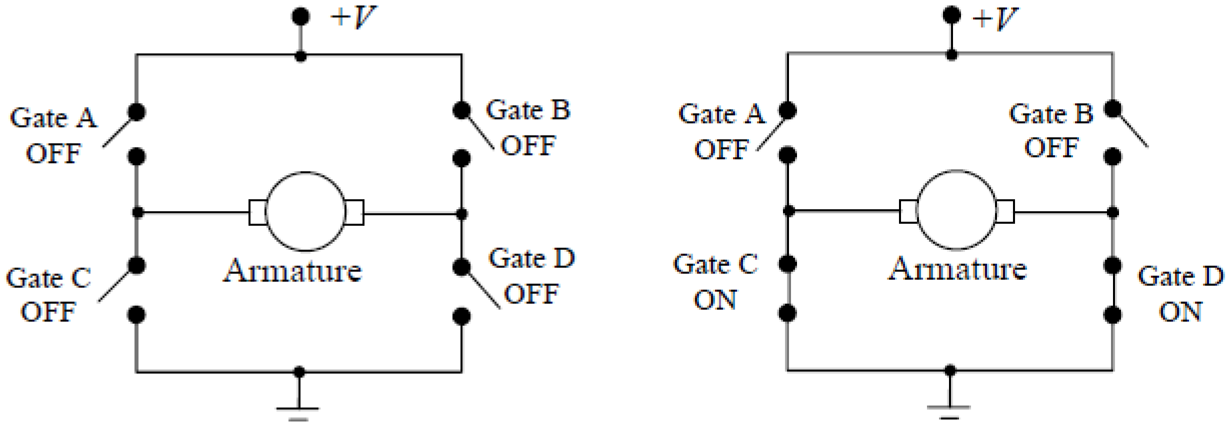
Figure 2.9: current to the motor is smoothed due to inductance

2.3.2 The H-bridge and bipolar PWM amplifiers

In most robotics applications, bi-directional control of motor speed is necessary. This requires a PWM amplifier to be bipolar, allowing for both forward and backward rotations. The architecture described in the previous section needs to be extended to meet this bipolar requirement. The H-Bridge architecture is commonly used for bipolar PWM amplifiers. As shown in Figure 2.10, the H-Bridge architecture resembles the letter H in the arrangement of switching transistors around the motor armature. Switching transistors A and B are pulled up to the supply voltage V, whereas transistors C and D are connected to ground. Combinations of these four switching transistors provide a variety of operations. In figure (i), gates A and D are ON, and B and C are OFF. This gate combination delivers a current to the armature in the forward direction. When the gate states are reversed, as shown in figure (ii), the direction of current is reversed. Furthermore, the motor coasts off when all the gates are turned OFF, since the armature is totally isolated or disconnected as shown in figure (iii). On the other hand, the armature windings are shortened, when both gates C and D are turned ON and A and B are turned OFF. See figure (iv). This shortened circuit provides a “braking” effect, when the motor rotor is rotating.



(i)Forward motion (ii) Reverse motion



(iii) The motor armature costs off (iv) The motor winding are shortened

Figure 2.10: H-bridge and four quadrant control

It should be noted that there is a fundamental danger in the H-bridge circuit. A direct short circuit can occur if the top and bottom switches connected to the same armature terminal are turned on at the same time. A catastrophic failure results when one of the switching transistors on the same vertical line in Figure 2.10 fails to turn off before the other turns on. Most of H-bridge power stages commercially available have several protection mechanisms to prevent the direct short circuit.

2.4 Optical Shaft Encoders

The servomechanism is based on analogue feedback technology, using a potentiometer and a tachometer generator. These analogue feedbacks, although simple, are no longer used in industrial robots and other industrial applications, due to limited reliability and performance. A potentiometer, for example, is poor in reliability, resolution, accuracy, and signal to noise ratio. The output tap of the variable resistance slides on a track of resistive material, making a mechanical contact all the time. This slide contact causes not only electric noise but also wear of the contacting surfaces. The resolution of the sensor are also limited by the mechanical contact. Furthermore, linearity depends on the uniformity of the resistive material coated on the substrate, and that is a limiting factor of a potentiometer's accuracy. Today's industrial standard is optical shaft encoders, having no sliding contact. This will be discussed next.

2.4.1 Basic principle

An optical encoder consists of a rotating disk with grids, light sources, photodetectors, and electronic circuits. As shown in Figure 2.11, a pattern of alternating opaque and translucent grids is printed on the rotating disk. A pair of light source and photodetector is placed on both sides of the rotating disk. As an opaque grid comes in, the light beam is blocked, while it is transmitted through the disk, when the translucent part comes in. The light beam is then detected by the photodetector. The disk is coupled to a motor shaft or a robot joint to measure. As it rotates, an alternating ON-OFF signal is obtained with the photodetector. The number of grids passing through the optical elements represents the distance traveled.

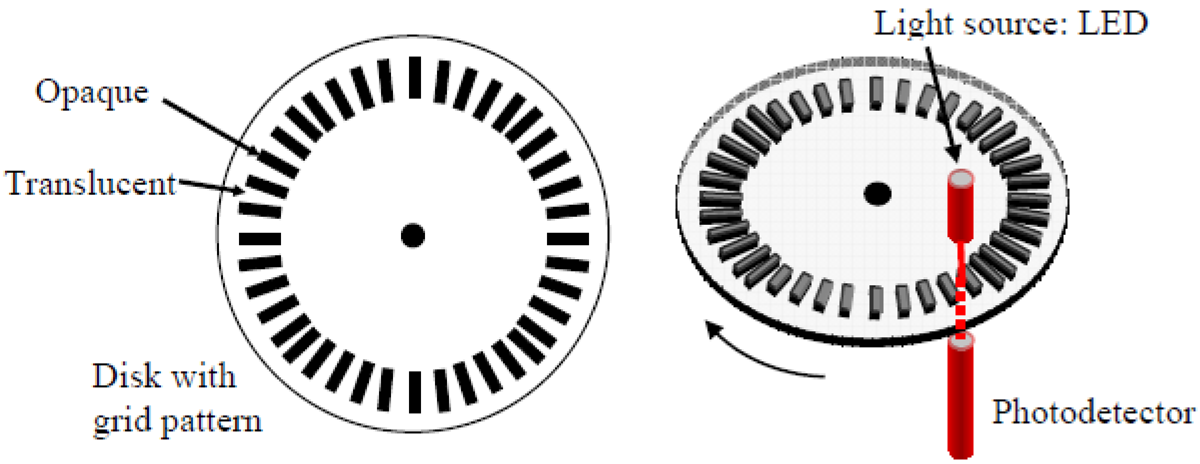


Figure 2.11: Basic construction of optical shaft encoder

This optical shaft encoder has no mechanical component making a slide contact, and has no component wear. An optical circuit is not disturbed by electric noise, and the photodetector output is a digital signal, which is more stable than an analogue signal. These make an optical shaft encoder reliable and robust; it is a suitable choice as a feedback sensor for servomotors.

2.4.2 Position measurement

One problem with the above optical encoder design is that the direction of rotation cannot be distinguished from the single photodetector output. The photodetector output is the same for both clockwise and counter-clockwise rotations. There is no indication as to which way the disk is rotating. Counting the pulse number merely gives the total distance the shaft has rotated back and forth. To measure the angular “position”, the direction of rotation must be distinguished.

One way of obtaining the directional information is to add another pair of light source/photodetector and a second track of opaque/translucent grids with 90 degrees of phase difference from the first track. Figure 2.12 illustrates a double track pattern and resultant output signals for clockwise and counter-clockwise rotations. Note that track A leads track B by 90 degrees for clockwise rotation and that track B leads track A for counter-clockwise rotation. By detecting the phase angle the direction of rotation can be distinguished, and this can be done easily with an up-down counter.

By simply feeding both A phase and B phase encoder signals to an up-down counter, the direction of rotation is first detected, and the number of rising edges and falling edges of both signals is counted in such a way that the counter adds the incoming edge number for clockwise rotation and subtract the edge numbers for counter-clockwise rotation. The up-down counter indicates the cumulative number of edges, that is, the angular “position” of the motor. The output of the up-down counter is binary *n*-bit signals ready to be sent to a digital controller withoutA/D (digital-to-analogue-conversion).

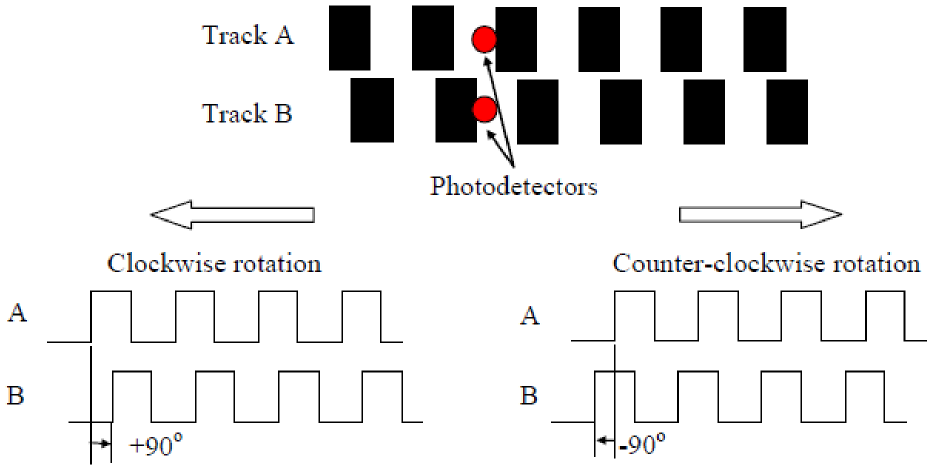


Figure 2.12: double track encode for detection of direction of rotation

It should be noted that this type of encoder requires initialization of the counter prior to actual measurement. Usually a robot is brought to a home position and the up-down counters are set to the initial state corresponding to the home position. This type of encoder is referred to as an ***incremental encoder***, since A-phase and B-phase signals provide relative displacements from an initial point. Whenever the power supply is shut down, the initialization must be performed for incremental encoders.

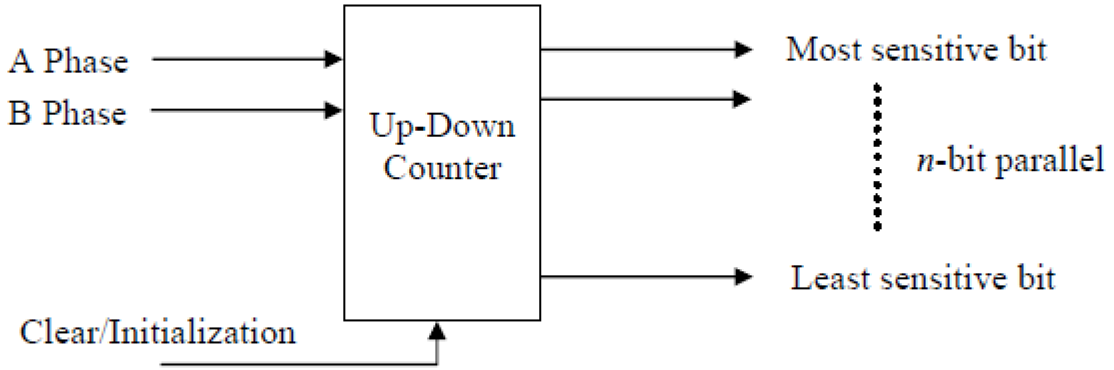


Figure 2.13: Up-down counters for an incremental encoder

2.5 Fluid Used In Actuators

2.5.1 Hydraulic Actuators

Hydraulic actuators are frequently used as joint or leg actuators in robotics applications requiring high payload lifting capability. Hydraulic actuators output mechanical motion through the control of incompressible fluid flow or pressure. Because incompressible fluid is used, these actuators are well suited for force, position, and velocity control. In addition, these actuators can be used to suspend a payload without significant power consumption. Another useful option when using hydraulics is that mechanical damping can be incorporated into the system design.

The primary components in a hydraulic actuation system include:

1. A pump: converts input electrical power to hydraulic pressure.
2. Valves: to control fluid direction, flow and pressure.
3. An actuators: converts fluid power into output mechanical energy.
4. Hoses or piping: used to transport fluids in the system.
5. Incompressible fluid: transfers power within the system.
6. Filters, accumulator, and reservoirs.

7. Sensors and controls.

Positive displacement pumps are used hydraulic actuator systems and include gear, rotary vane, and piston pumps. The valves that are used include directional valves (also called distributors), on-off or check valves, pressure regulator valves, flow regulator valves, and proportional or servo valves.

Both linear and rotary hydraulic actuators have been developed to convert fluid power into output motion. A linear actuator is based on a rod connected to a piston which slides inside of a cylinder. The rod is connected to the mechanical load in motion. The cylinder may be single or double action. A single action cylinder can apply force in only one direction and makes use of a spring or external load to return the piston to its nominal position. A double action cylinder can be controlled to apply force in two directions. In this case, the hydraulic fluid is applied to both faces of the piston.

Hydraulic actuators have been used in many factory automation problems and have also been used in mobile robotics. Figure 2.14 is a picture of the TITAN 3 servo-hydraulic manipulator system from Schilling Robotics. This is a remote manipulator that was originally developed for mobile underwater applications but is also being used in the nuclear industry.

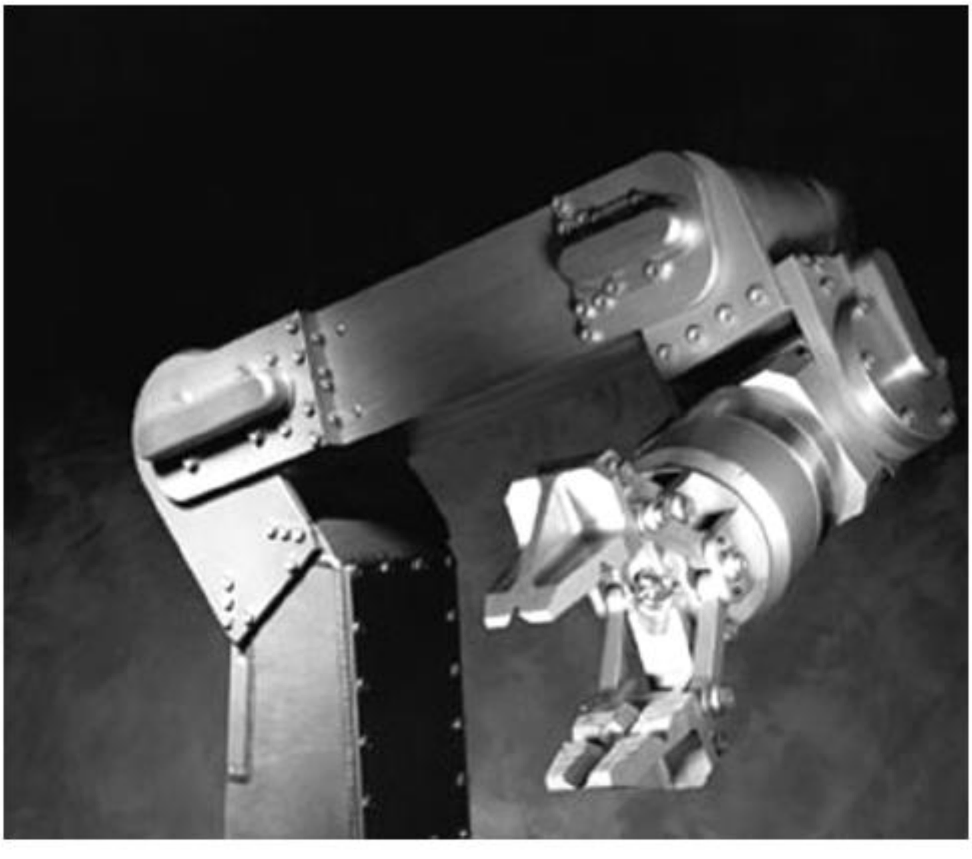


Figure 2.14: Titan 3 servo-hydraulic manipulator

2.5.2 Pneumatic Actuators

Pneumatic actuators are similar to hydraulic actuators in that they are also fluid powered. The difference is that a compressible fluid, pressurized air, is used to generate output mechanical motion. Pneumatic actuators have less load carrying capability than hydraulic actuators because they have lower working pressure. However, pneumatic actuators have advantages in lower system weight and relative size. They are also less complex in part because exhausted pressurized air in the actuator can be released to the environment through an outlet valve rather than sent through a return line.

Because compressed air is used, the governing dynamic equations of pneumatic actuators are nonlinear. In addition, compressed air adds passive compliance to the actuator. These two factors make these actuators more difficult to use for force, position, and velocity control. However, pneumatic actuators are frequently used in industry for discrete devices such as grippers on robotic end effectors.

The primary components in a pneumatic actuation system include:

1. A compressor: converts input electrical power to air pressure
2. Compressed air treatment unit: includes filters and pressure regulation and lubricators.
3. Valves: to control pneumatic power.
4. An actuator: converts pneumatic power into output mechanical power.
5. Hoses or piping: used to transport the air in the system.
6. Sensors and controls.

There are many types of pump technologies used in pneumatic compressors. They include positive displacement pumps such as piston, diaphragm, and rotary vane types as well as non-positive displacement pumps such as centrifugal, axial, and regenerative blowers. The compressor may include a storage tank or it may output pressurized air directly to a regulator valve. The types of valves used are similar to those used in hydraulic actuation systems as described in the previous section. Both rotary and linear actuators are available and are also similar in design to those used in hydraulic actuation systems.

2.6 Common Uses for Digital Sensors

Digital sensors can be used in a wide variety of applications within robotics. These include proximity sensors, limit sensors, and safety sensors such as light curtains.

2.6.1 Proximity Sensors

Proximity sensors are similar to analog displacement sensors, but they offer a static digital output as opposed to an analog output. Proximity sensors are used to determine the presence or absence of an object. They may be used as limit sensors, counting devices, or discrete positioning sensors. They are typically noncontact digital sensors and are based on inductive, capacitive, photoelectric, or Hall effect technology. These technologies are discussed in the previous section on analog sensors. Their design is frequently similar to that of analog position sensors but with threshold detecting electronics included so that their output is digital.

2.6.2 Limit Switches and Sensors

Limit switches or limit sensors are digital inputs to a robot controller that signal the end of travel for motors, actuators, or other mechanisms. The incorporation of limit of limit sensors help. Prevent mechanical failure caused by part of a mechanism hitting a hard stop in the system. The limit sensor itself can be a physical switch with mechanical contacts or a digital proximity sensor as described above. Limit sensors may be mounted to individual joints in the robot or to axes of motion in a robotic workcell. When the limit sensor is encountered for a particular joint or axis, the robot controller will bring the motion to a safe stop.

Both a forward and a reverse limit sensor can be connected to each joint or axis in a robotic system. Forward is defined as the direction of increasing position as measured by the encoder or analog feedback signal. Limit sensors can be used with both rotational and linear axes. A home switch or sensor can also be built into each axis and used to indicate a center position reference for the axis.

.

2.6.3 Light Curtains

Light curtains can automatically detect when an operator moves within the danger area for a robot or robotic operation. This danger area will usually include the entire workspace of the robot. Light curtains are typically based on photoelectric sensors which emit multiple beams of infrared light. When any of the beams of light is broken, the control circuit for the light curtain is activated, and the robotic system is immediately shut down in a safe manner.

2.7 Vision

Many robots use industrial cameras for part detection, inspection and, sometimes, guidance. The camera output may be analog or digital and may be acquired by a computer through several means. Often, a frame grabber or image acquisition plug-in board is used and installed in a computer. More recently, computer bus technologies such as IEEE 1394, Camera Link®, Gigabit Ethernet, and USB have been used to transfer data between the camera and the computer.

2.8 Batteries

Robots may be powered by a variety of methods. Some large robots use internal combustion engines to generate electricity or power hydraulic or pneumatic actuators. For a small robot, however, battery power offers a number of advantages over any other method. Batteries are cheap, relatively safe, small, and easy to use. Also, motors convert electrical power into mechanical power with relative efficiency. There are many different types of batteries, each with its own tradeoffs. This section introduces a variety of batteries, explains standard ways of rating batteries, and discusses the design of the battery charger.

2.8.1 Cell Characteristics

Two terms that are often used interchangeably, but actually have a different meaning, are the words *battery* and *cell*. Technically, a cell is a unit that houses a single chemical reaction to produce electricity. A battery is a bank of cells.

2.8.1.1 Voltage

Cells use chemical reactions to produce electricity. Depending on what materials are used to create the reaction, a different voltage will be produced. This voltage called *a nominal cell voltage* and is different for different battery technologies.

For example, a standard flash light cell uses a carbon-zinc reaction and has a cell voltage of 1.5 volt. Car batteries have six lead-acid cells each with a cell voltage of 2 volts (yielding the 12 volt battery).

2.8.1.2 Capacity

In general, the larger the cell is, the more electricity it can supply. This *cell capacity* is measured in *ampere- hours*, Which are the number of hours that the cell can supply a certain amount of current before it's voltage drops below a predetermined threshold value.

For example, 9 volt alkaline batteries (which consist internally ofsix 1.5 volt alkaline cells) are generally rated at about 1 ampere hour. This means that the battery can continuously supply one ampere of current for one hour before \dying." In the capacity measurement, the 9 volt alkaline battery \dies" when the battery voltage drops below 5.4 volts.

2.8.1.3 Power density

There are large differences in capacity per unit weight-the cell's power density- across battery types. This is of the cell most important rating.

Inexpensive carbon-zinc cells have the lowest power density of all cell types. Alkaline cells have about ten times the power density of carbon-zinc cells. Nickel-cadmium cells have less power density than alkalines, but they are rechargeable.

2.8.1.4 Internal resistance

A cell can be modelled as a perfect voltage source in series with a resistor. When current is drawn out of the cell, its output voltage drops as voltage is lost across the resistor.

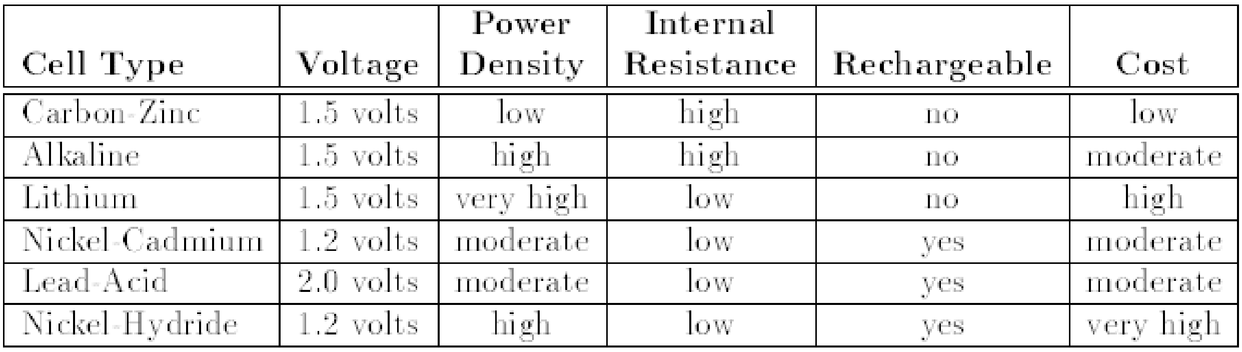
This cell characteristic, called the *internal resistance*, is important because it determines the maximum rate at which power can be drawn out the cell.

2.8.1.5 Rechargeability

Another important characteristic of a cell is whether or not it is rechargeable, and if so, how many times. Because cells are quite toxic to the environment, use of rechargeable cells is an important issue.

Unfortunately, the cells with highest power densities-alkaline and lithium- aren't rechargeable. But advances in rechargeable technologies are catching up.

Table 2.1: cell characteristic



Cell Type	Voltage	Power Density	Internal Resistance	Rechargeable	Cost
Carbon-Zinc	1.5 volts	low	high	no	low
Alkaline	1.5 volts	high	high	no	moderate
Lithium	1.5 volts	very high	low	no	high
Nickel-Cadmium	1.2 volts	moderate	low	yes	moderate
Lead-Acid	2.0 volts	moderate	low	yes	moderate
Nickel-Hydride	1.2 volts	high	low	yes	very high

2.8.2 Battery packs

There are two ways that cells may be combined to make batteries: series connections and parallel connections.

When cells are connected in series, their voltages add but their amp-hour capacity does not. Series batteries should be composed of cells of equal capacities. When cells are connected in parallel, their voltages remain the same, but their capacities add.

2.9 Robot programming languages

In this section we began to consider the interface between the human user and the robot. It’s by means of this interface that a user takes advantage of all the underlying mechanics and control algorithms.

2.9.1 The three levels of robot programming

There have been many styles of user interface developed for programming robots. Before the rapid proliferation of microprocessor in industry, robot controllers resembled the simple sequencers often used to control fixed automation. Modern approaches focus on computer programming, and issues in programming robots include all issues faced in general computer programming, and more.

1. Teach by showing

Early robots were all programmed by a method that we will call *teach by showing*, which involved moving the robot to a desired goal point and recording it’s position in a memory which the sequencer would read during playback.

2. Explicit robot programming languages

Robot programming languages have taken on many forms as well. We will split them into three categories as follows:

1- Specialized manipulation languages.

2- Robot library for an existing computer language.

3- Robot library for a new general-purpose language.

3. Task level programming languages

The third level of robot programming methodology is embodied in *Task level programming languages*. These are languages which allow the user to command desired sub goals of the task directly, rather than to specify the details of every action the robot is to take.

2.9.2 A sample application

Figure 2.16 shows an automated work cell which consists of a conveyor under computer control which deliver a workpiece. A camera connected to a vision system is used to allocate the workpiece on the conveyor. There is an industrial robot (i.e. a puma 560 is pictured) equipped with a force sensing wrist.

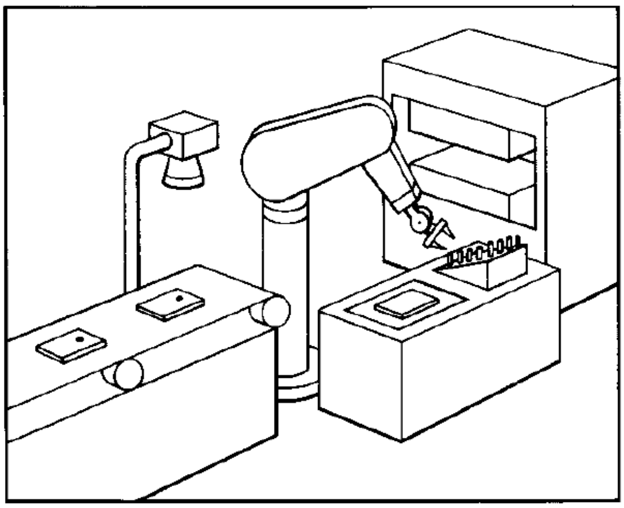


Figure 2.15: an automated work cell containing an industrial robot

2.10 Literature Reviews

It has been designed in Palestine polytechnic university (PPU), a robot for painting, the designers use it to replace the role of man in painting process, and with helping to get rid of most hazards in painting process, they use the Cartesian robot that facilitates the painting process and give a fixed distance from the painted plate, the data about the plat are inserted by the personal, and if there is error occurred while inputting data inside the computer about the measurement of the plate the source of painting will be turned off using photo sensor.

In their system they used pneumatic elements which represent in reciprocating compressor, spray gun, rubber pipes, and the quantity of painting can be controlled by nozzle at the top of the spray gun.

Chapter three

System design and simulation

System design and simulation

3.1 Structure

In this chapter we try to talk about the component which will use in designing our robot, and the items used in building our robot frame, containing the beams, bricks, gears, sensors, motors, wheels…etc. beside to all of the that the brain of the robot which represent in the NXT kit, and type of programming used in our robot. Here in the figure 3.1 shown, the flowchart obviously shows you the function of the robot, and how it works with the program downloaded on the kit “robot brain”.

The block diagram in Figure 3.2 also show the component of the robot and its functions, the figure show how the rechargeable battery supply the NXT kit with 9V, which is embedded with a Microcontroller which needs a 5V to operate so that the kit has its own resistance to make drop voltage for proper operating of the Microcontroller.

The sensors that work on 5V, should be connected to the Input ports of the Microcontroller, each sensor has 3 wires, two of them for the power supply and the other one for the electrical signal that which has been changed from physical quantity to electrical quantity, the wires that connect them together called RJ12, we will use for about 5 sensors “sound, light, touch, ultrasonic and rotational sensor (encoder)” each sensor has his own function to do.

The outputs are the motors which are connected to the output ports of the Microcontroller, we have three outputs, and all of them are DC-Servo motors, these motors have a rotational sensor “encoder” embedded in, the motors work on 9V power supply and two of them have 6 wires while the third not, the cable which will connect them called RJ12, two wires for the power supply for the motor and another tow for the power supply of the motor while the last two, one of them to measure the distance in the forwarding, while the last one for measure the distance in backward mode, you will see the motors simulated on the MATLAB, the speed is controlled through the PWM of the H-bridge converter with assistant of the gears and wheels.

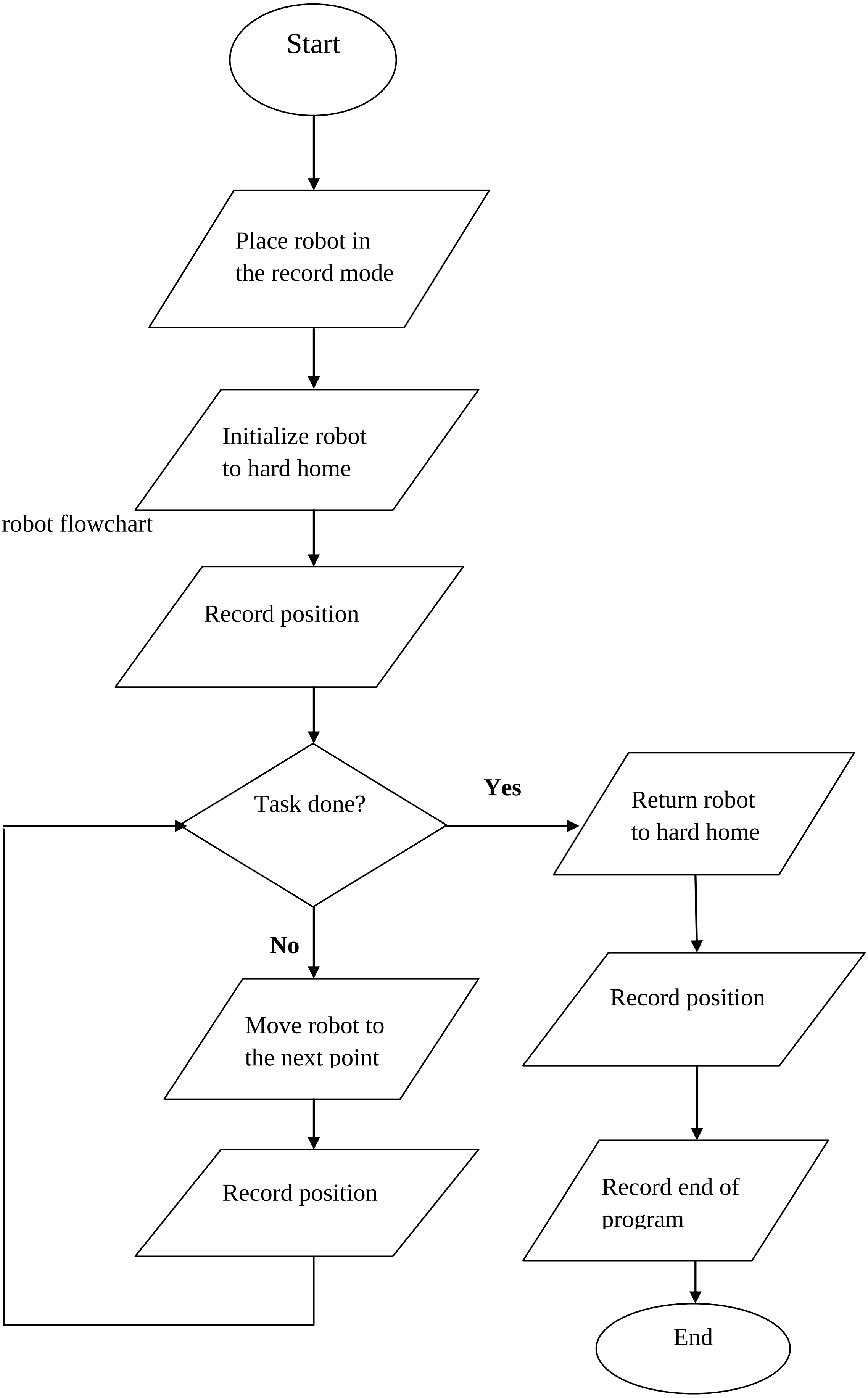


Figure 3.1: robot flowchart

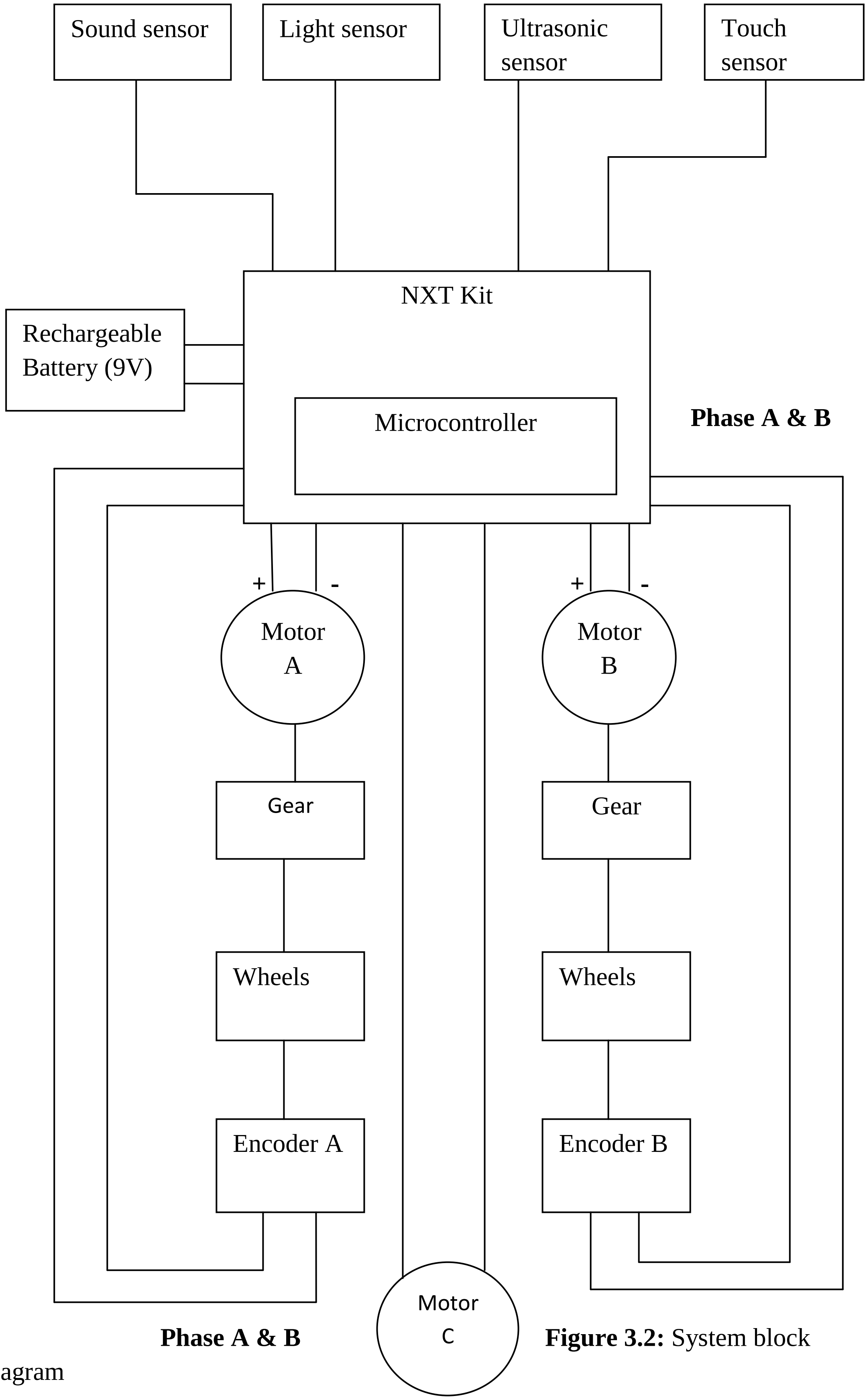


Figure 3.2: System block diagram

3.1.1 Bricks, plates, and beams

Bricks, plates, and beams are not as glamorous as the NXT brick, motors, and sensors. But they are the fundamental components that are used to build the frame that supports the NXT, counteracts the motor's forces, and holds the sensors precisely in place. Master some robot fundamental frames, which give the success for the project, ignoring them will force you to spend more time for repairing your robot than you build it.

3.1.1.1 Bricks

Bricks are made out of ABS plastic. They are injection molded to very exacting tolerances (0.002mm)[1]. The top of the brick is covered with cylindrical plastic bumps called studs. The bottom of the brick has cylindrical holes or tubes. When you snap two bricks together, the tubes deform slightly around the studs, locking the two firmly together.

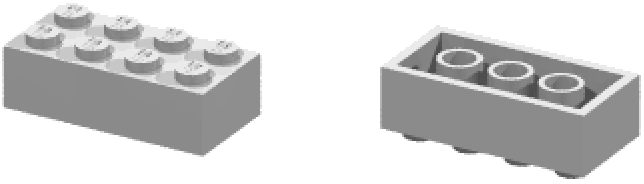


Figure 3.3: Plastic bricks

3.1.1.1.1 Dimensions

The common practice is to refer to bricks using their dimensions: width, length, and height (though height is often left off when referring to standard sized bricks). When doing this, the width and length dimensions are given in studs. The piece below is a 2 x 4 brick. Bricks are based on the metric system. The 2 x 4 brick above is 16mm wide, 32 mm long, and 9.6mm high (ignoring the studs on top). That works out to 1 stud = 8mm. It also means that bricks are 1.2 studs high. This asymmetry can lead to design and building difficulties as will be discussed later.

---

[1] From "Jin Sato's Lego Mindstorms The Master's Technique"

3.1.1.2 Plates

Plates are essentially short bricks. They are 1/3 the height of standard bricks--3.2mm or 0.4 studs. Plates use the same naming convention as bricks. Some plates have through holes aligned with the backside tubes. They are referred to as Technical plates, or less obscurely, plates with holes. The holes accept axles and connector pins and make the Technical plates much more useful.

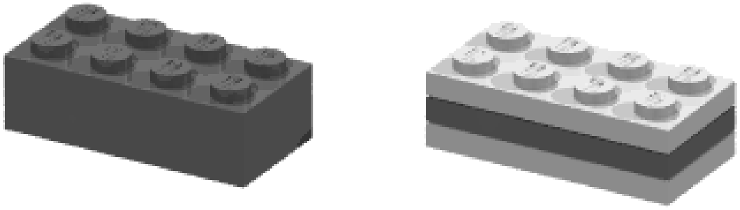


Figure 3.4: three plates = one brick high

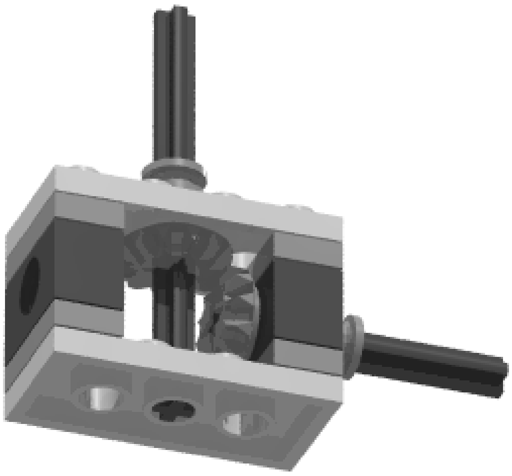


Figure 3.5: simple gear box using Technical plates

3.1.1.3 Beams

A series of complex models for older children to build. Central to Technical are the new beams which are 1x bricks with holes in their sides. The holes are spaced at one-stud intervals and centered between the studs on the top of the beam. The beams can be stacked on top of each other just like bricks. In addition, connector pins can be placed in the side holes allowing the beams to be assembled side by side. The number of assembly techniques available using the new parts is staggering.

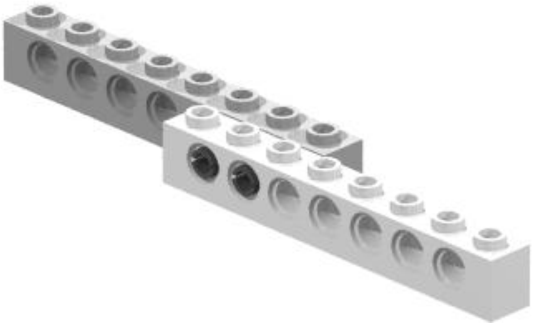


Figure 3.6: technical beams

3.2 Building frame

A robot needs some sort of frame. The frame gives the robot its shape. It provides mounting points for sensors and reacts the forces generated by motors and gears. It is like our skeleton, which gives us our shape, supports our organs, and reacts the forces generated by our muscles. A good frame is strong, lightweight, and holds together even after much use.

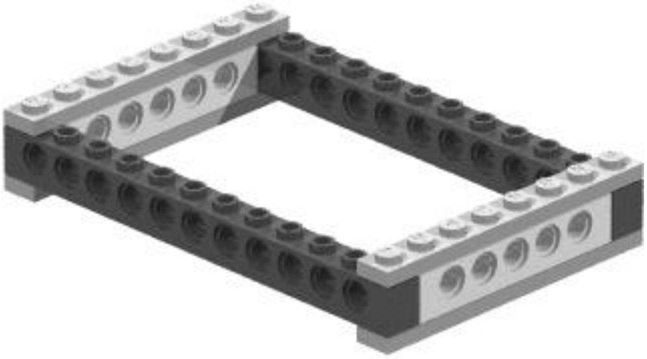


Figure 3.7: simple frame

Figure 3.7 shows a simple frame made out of beams and 1x8 plates. It's strong, lightweight, and the dimensions are appropriate for the base of a robot platform. But it is not very rigid. A gentle push on opposing corners causes the frame to twist out of shape. Eventually the corner connections work loose, and the frame falls apart.

The problem is that the plates do not lock the corners at right angles. There is a small amount of clearance between the ends of the 1 x 6 beams and the sides of the 1 x 12 beams. This allows the studs to act as hinges. Replacing one or more of the 1 x 8 plates with 2 x 8 plates makes the frame much more rigid.

This same technique can be used to build tall frames that are lightweight and strong. Figure 3.8 shows a tall frame that uses friction pins at the corners to attach the vertical and horizontal members. Notice that the 1-2-1 (1 Beam-2 Plates-1 Beam) technique is used to get the proper spacing.

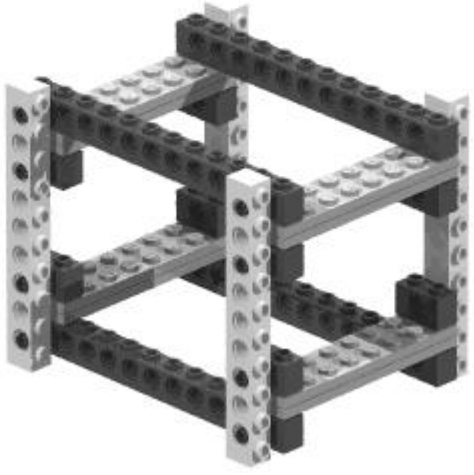


Figure 3.8: A Tall frame

3.2.1 Diagonal bracing

It's very easy to get locked into the mindset that horizontal and vertical are the only ways to build. The shape looks like grid. But diagonal connections are possible as well. Diagonal bracing is trickier to implement than perpendicular cross bracing. Cross bracing can be used on any assembly where the dimension is evenly divisible by two studs, but it can't be used anywhere else. With diagonal bracing there are more solutions, but their derivations are not as obvious.

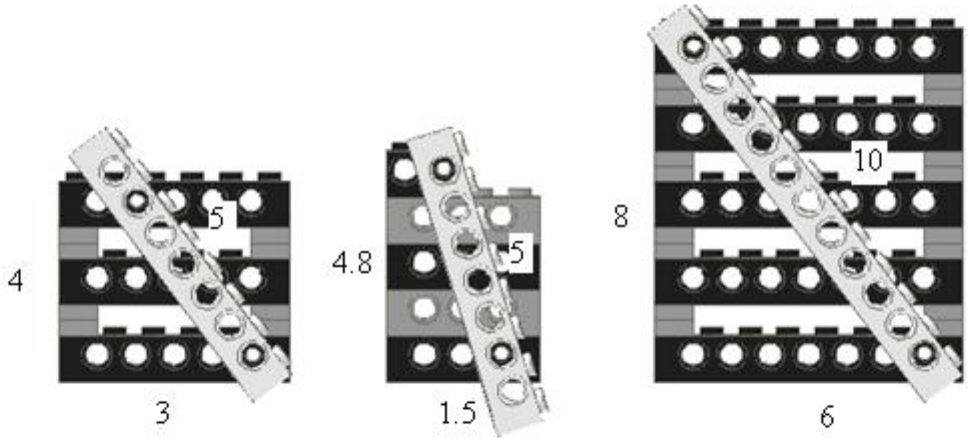


Figure 3.9: diagonal cross bracing

You can find diagonal bracing solutions through experimentation. Lay the diagonal brace on the part, and move it about until you find a place where the holes line up. It can also be done analytically using the Pythagorean Theorem for right triangles. “The sum of the squares of the legs of a right triangle equals the square of the hypotenuse.” This is often written as “$C = \sqrt{A^2 + B^2}$.” The two legs are the base (width across the bottom) and the height. The hypotenuse is the diagonal beam. Diagonal bracing is possible if the hypotenuse is close to an integer number (less than 0.05studs difference).

3.3 Gears

Eventually you will want to make your robot move. The 9 volt motors provide the motive power, but they may not run at the right speed or be powerful enough. It may also be too difficult to position the motors where they can be directly attached to the wheels. All these problems can be solved using gears.

Gears are generally used for one of the following reasons:

1. to transmit torque from one angle to another.
2. to increase and decrease the speed of rotation.
3. To reverse the direction of rotation.
4. To move rotational motion to a different axis.
5. To change rotary motion to linear motion.
6. To keep the rotation of tow axles synchronized.

3.3.1 Spur gears

A spur gear is used when shafts must rotate in the same plane. In a spur gear the teeth are straight and parallel to the shaft. They are by far the most common type of gears, and they are what most people picture when you mention gears. Our robot includes four different sized spur gears in the Robotics Invention System.

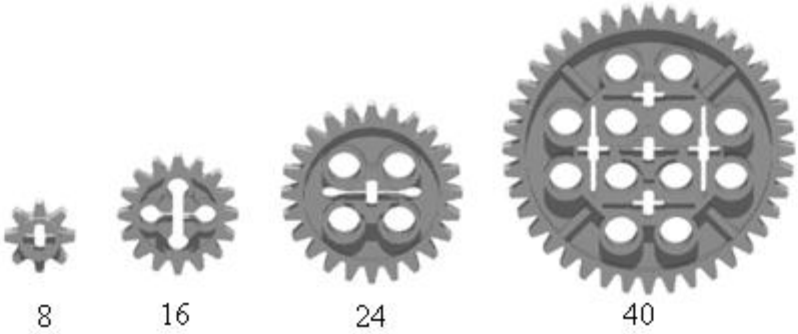


Figure 3.10: spur gears

Gears are normally referred to by their type and the number of teeth. Take, for example, an 8 tooth spur gear. Sometimes a kind of shorthand notation is used where “tooth” is replaced with “t” and the type is not specified for spur gears (because they are so common). For example, a 40 tooth spur gear would be referred to as a 40t gear.

3.3.1.1 Gear spacing

Sizes of spur gears are shown in table 3.1 It is interesting to note that the ratio of the radii is equal to the ratio of the tooth count (8/24 = 0.5/1.5 = 1/3). This is because all the different sized spur gears have the same sized teeth--even the little 8t gear with its involutes profile gear teeth. Having the same sized teeth allows the gears to mesh properly.

Table 3.1: spur gear sizes

Teeth	8	16	24	40
Radius (studs)	0.5	1	1.5	2.5

Knowing the radius of a gear and the number of teeth, we can calculate the size of each tooth. This information can be useful to know when using a spur gear with a gear rack.

Circumference = 2 x Pi x Radius (3.1)

Circumference = Tooth Size x Tooth Count (3.2)

Size x Count = 2 x Pi x Radius (3.3)

Size = 2 x Pi x Radius / Count (3.4)

16 tooth = 2 x Pi x 1 / 16

= Pi / 8 studs

= 0.392 studs or 3.14 mm

Check it out for yourself. The teeth for each spur gear really do evaluate to the same size! There are three different types or shapes of spacing the gears such that the horizontal spacing, vertical spacing, diagonal spacing, which they are presented in the figures below

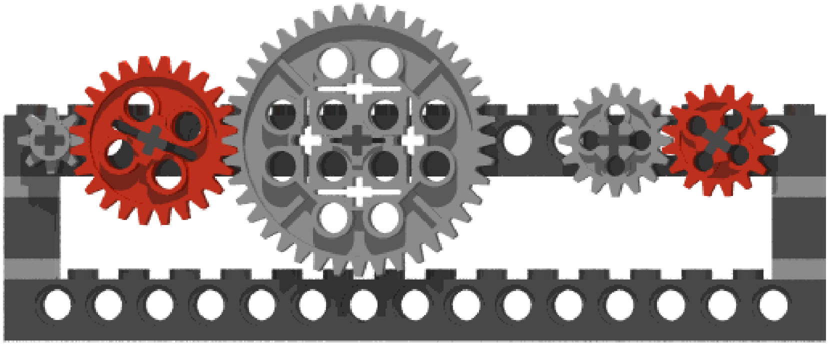


Figure 3.11: Horizontal spacing

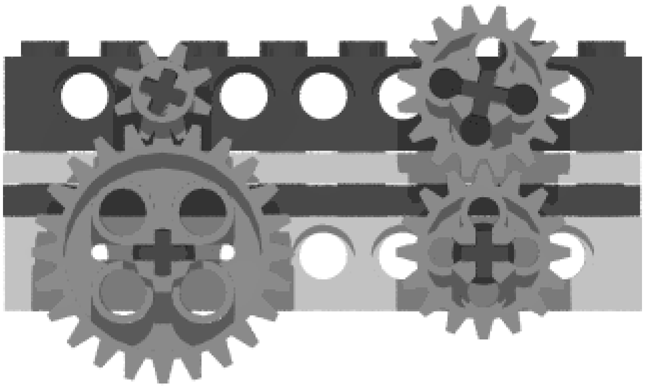


Figure 3.12: Vertical spacing

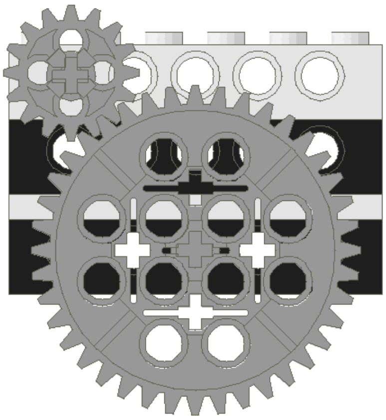


Figure 3.13: diagonal spacing

3.3.2 Gear ratio

Gear ratio is how much the output shaft of a gearbox turns for a given rotation of the input shaft. In figure 3.14, we have a gearbox consisting of two gears: an 8t gear on the input shaft and a 24t gear on the output shaft. If the 8t gear rotates one full revolution then eight of its teeth would pass through the starting line. Because the two gears are meshed, eight of the 24t gear's teeth would also pass the starting line. Since the teeth are evenly distributed around the circumference of the gear, the 24t gear turns 8/24ths or 1/3 of a revolution.

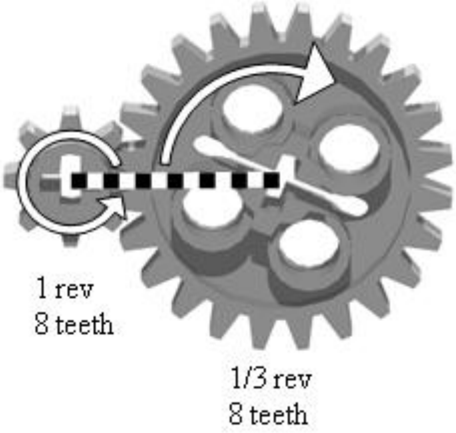


Figure 3.14: Gear ratio

Using the rotation information we can calculate the gear ratio. Following popular convention, it is expressed as a ratio of whole numbers.

Gear ratio = 1 : 1/3
= 3 : 1

The 3:1 gear ratio tells us that the input shaft (attached to the 8t gear) has to complete three full revolutions for the output shaft (attached to the 24t gear) to rotate all the way around just once. Using gears to slow down rate of rotation or decrease the amount of rotation is called gearing down. If we were to switch the 8t and 24t gears around, the output shaft would spin three revolutions for each revolution of the input shaft. This is gearing up, and the gear ratio would be 1:3.

You may have noticed that the gear ratio is the inverse of the ratio of the number of gear teeth. The reason for this is easier to see if we recalculate the gear ratio using an input shaft rotation of only 1 tooth. In the case of the 8t gear driving the 24t gear, the input shaft would turn 1/8 of a revolution and the output shaft 1/24 of a revolution.

3.3.3 Torque

Torque is a force that tends to rotate or turn things. You generate a torque any time you apply a force to the handle of a wrench. This force creates a torque on the nut, which tends to turn the nut. If the nut is too tight, you either pull harder (more force), or get a longer handled wrench (more distance). From the wrench example, we see that torque is a product of force and distance. The units used when measuring torque reflect this. In the U. S. we measure torque in foot-pounds (ft-lbs). Newton-meters (Nm) is the unit of torque in the metric system.

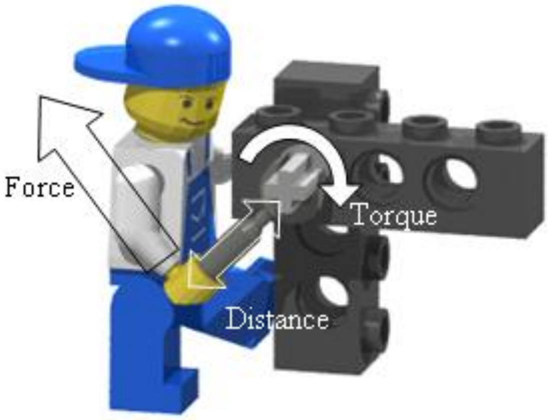


Figure 3.15: Torque relation with the distance and force

Gears operate by transmitting forces at the teeth of the gear. In the 24t gear is generating a force against the teeth of the 40t gear. The force (f) is equal to the torque (t1) applied to the 24t gear divided by the radius (r1). The force transmitted by a gear is inversely proportional to the gear's radius. The larger the radius of the gear, the less force it will generate for a given torque.

From the discussion earlier, we know that the gear ratio is the inverse of the ratio of the gears' radii. This allows us to use the gear ratio to calculate the torque amplification of a gear system. Using the gear ratio for the example above:

$$\text{Gear ratio} = r2 : r1 \tag{3.5}$$

$$= r2 : r1$$

$$= 5 / 3$$

$$t2 = t1 * r2 / r1$$

$$= t1 * \text{gear ratio}$$

3.3.4 Speed

When using simple machines like gears (or any kind of machine for that matter) you never get something for nothing. In our prior example, we used gears to increase torque. What we traded to get the torque increase was speed. The output shaft may turn stronger, but it also turns slower.

If we measure the angles in radians (1 radian = 180 degrees / Pi = 1 revolution / (2 x Pi)), the tooth velocity of a gear (v) is equal to the angular velocity (ω) times the radius (r). There is a proportional relationship between the radius and the tooth velocity. At a given angular velocity, the teeth of a larger gear will travel faster than the teeth of a smaller gear.

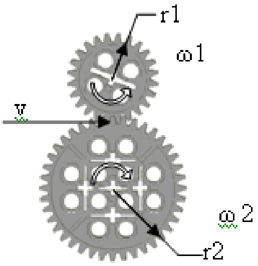


Figure 3.16: the relation between the speed and the ratio

$$V = \omega 1 \times r1 \tag{3.7}$$

$$\omega 2 = v / r2 \tag{3.8}$$

$$\omega 2 = (\omega 1 \times r1) / r2 \tag{3.9}$$

$$\omega 2 / \omega 1 = r1 / r2. \tag{3.10}$$

The larger gear spins more slowly than the smaller gear. The ratio of the angular velocities is equal to the inverse of the ratio of the radii. Knowing this we can calculate the angular velocity using the radius ratio directly.

3.3.5 Backslash

The backlash of a gear train is the amount the input shaft can rotate without moving the output shaft. Backlash is caused by the gears not meshing perfectly Figure 3.17. In this example, when gear A reverses rotation, the tooth on gear B goes from being loaded on the left side to being loaded on the right side. Because of the gap between the teeth, A will be able to rotate a small amount before B notices the change in direction.

Backlash introduces discontinuity, uncertainty, and impact in mechanical systems. This makes accurate control difficult. Positioning accuracy is also compromised due to backlash. A large amount of backlash makes a robot feel sloppy and gives an overall impression of poor engineering (you want to impress those judges).

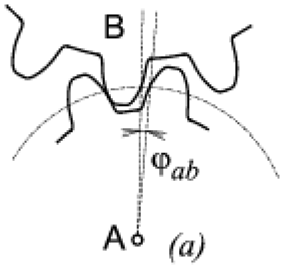


Figure 3.17: Backslash caused by poor meshing

In industry, backlash is reduced by using specially mated gears that are designed to mesh perfectly. Gears are designed to work with a wide variety of gears of different sizes and designs. This limits how perfectly they can fit together. Luckily, there are other ways to minimize backlash and its effects.

Reducing backslash:

1. Place rotation sensor near the output shat, this minimizes the effect backslash has on the sensor readings.
2. Use large gears. Backslash is less noticeable in larger gears.
3. Minimize the number of gears in a gear train, the larger the number of meshes, the greater the backslash.
4. The backslash increases when you gear up and decreases when you gear down.

An interesting alternative is to prevent backlash from occurring. You can take the play out of a gear train by applying a torque to the output shaft, trading some input torque for more precision. If the torque is great enough, it prevents backlash from happening when changing direction.

3.4 Pulleys

A pulley is a wheel with a groove about its diameter. The groove, called the race, accepts a belt which attaches the pulley to other pulleys. As the pulley rotates, friction forces pull on the belt putting it in tension. The belt transmits the force to the other pulley causing it to rotate.

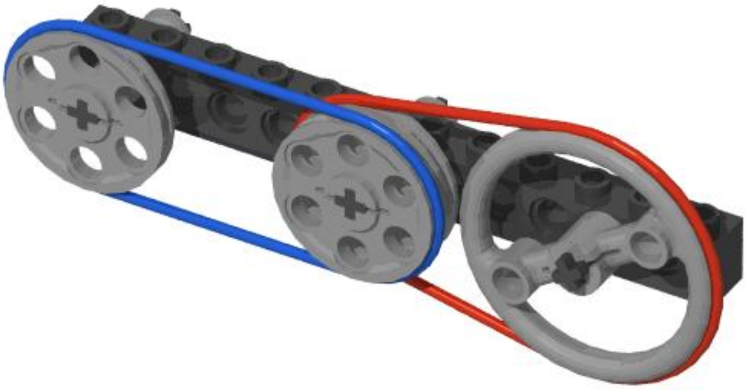


Figure 3.18: Pulleys and belts

Look to these four sizes of pulleys below, it's possible to "gear up" and "gear down", the reduction ratio of pulleys are determined by the ratio between their diameters. The trick is determining exactly where to measure the diameters

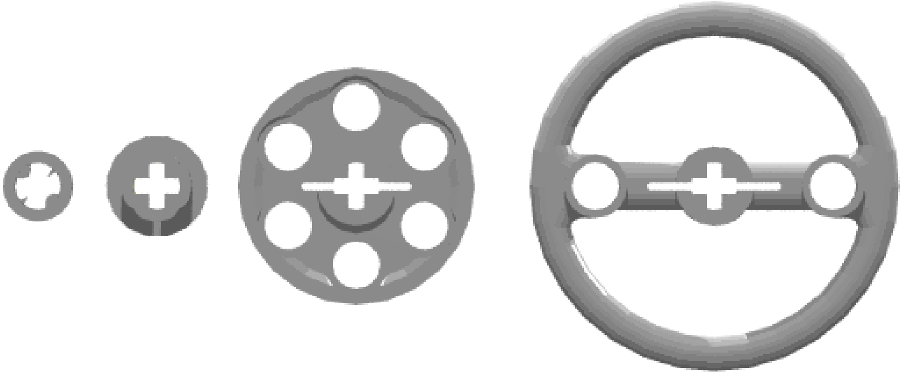


Figure 3.19: Pulleys with different diameters.

3.5 Wheels

Challenges are involved moving around and picking up things, moving around and dropping things, moving around and triggering things. All this moving around requires some sort of mobile platform, and most of the mobile platforms use wheels.

Wheels support the weight of the robot and transmit the power of the motors to the ground. What wheels you use will affect your robot's speed, power, accuracy, and ability to handle variations in terrain. Wheel choices will have a profound effect on your robot's success or failure.

3.5.1 Sizes

There are three sizes of solid rubber tires which all fit on the same plastic wheel and three sizes of balloon tires that fit on differently sized hubs. Dimensions in millimeters are stamped on the sidewall of the balloon tires. Dimensions for the solid tires are not available; therefore, approximate dimensions are supplied in the figure below.

Figure 3.20: robot wheel sizes

The diameter of the wheels chosen will, in a large part, determine how fast and powerful your robot is. Large wheels will make a robot run faster but decrease its towing capacity. Small wheels will give you more power but with a corresponding decrease in speed.

$$\text{Circumference} = \text{Pi} * \text{diameter} \qquad (3.11)$$

$$v = \omega \times \text{Pi} \times d \times a \qquad (3.12)$$

ω: the angular velocity, d: the diameter of the wheel, v: the linear velocity, a: gear ratio.

3.6 Balance

Proper balance is very important in robot design. For your robot to move in a predictable and repeatable manner, all wheels must be in contact with the ground at all times, and the weight carried by each wheel must be consistent. Improper balance will result in a robot that tips over easily or lifts its wheels when accelerating or turning.

Balance is dependent upon two factors: wheel base and center of gravity. Center of gravity (CG) is the point within your robot where there are equal amounts of mass above and below, equal amounts to the left and right, and equal amounts fore and aft. For stability calculations, we pretend that all the mass of your robot is concentrated in this tiny spot. The wheel base is the region outlined when you play connects the dots with the points where your robot's wheels touch the ground.

3.6.1 Inertia

Determining the vertical component of the center of gravity is more difficult than finding the left/right center or the fore/aft center. So why bother doing it? After all, if you know that the CG (central of gravity) is near the center of the wheelbase, then the robot is guaranteed to be stable, right? Well, not always. Figure 3.21 shows how climbing an incline moves the effective CG. The robot with the lower CG is still stable, but climbing the incline moves the CG behind the rear wheels on the robot with the higher CG, causing it to tip over backwards.

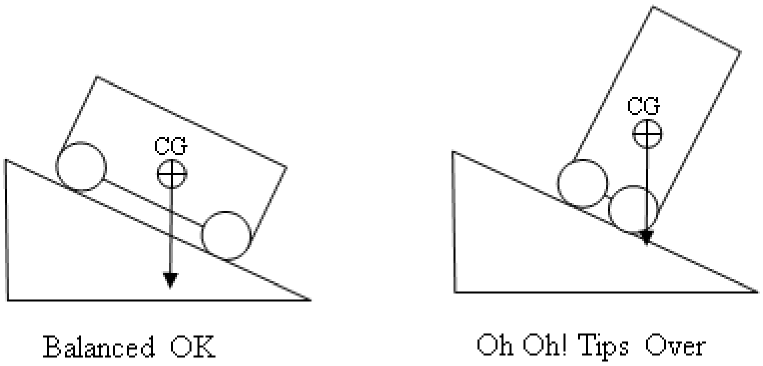


Figure 3.21: an incline moves to effective CG

3.6.2 Finding the center of gravity

Determining your robot's CG is an informative and educational activity. There are many ways this can be done, ranging from the dry and boring (summing the moment of inertia for each component and dividing by the total mass) to the slightly frightening (hanging the robot from a string). A fairly safe and easy way to locate the CG is the balance method.

To determine the CG using the balance method you need to locate the balance point on the lateral, longitudinal, and vertical axes. Each balance point defines a plane upon which the CG resides. The CG is located at the intersection of the three planes.

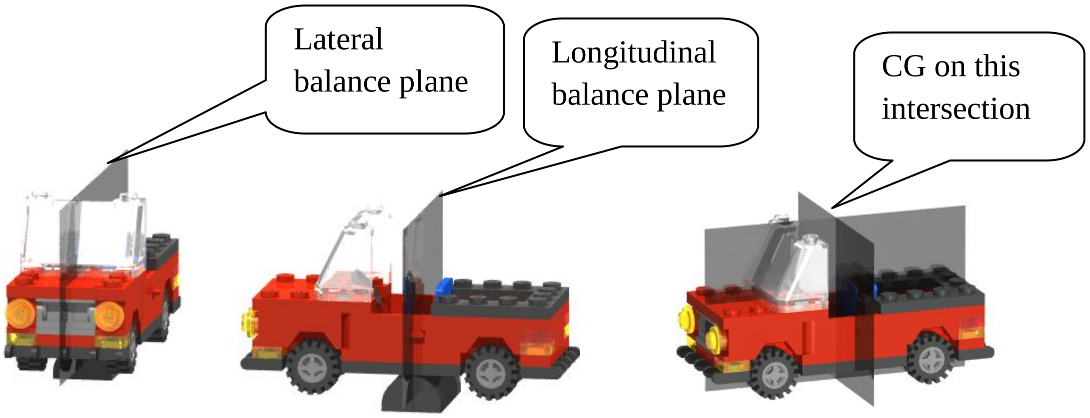


Figure 3.22: finding CG using balance method

To find the balance points, you need a fulcrum that will support the weight of the robot while still allowing it to tip easily. Look to the intersection is the fulcrum in the figure above. Place the robot on the fulcrum such that the fulcrum is parallel to the balance plane you are trying to locate. Slowly adjust the position of the fulcrum until the robot balances. This is the balance point. Repeat for the other two axes to find the CG.

3.7 Robot drive

3.7.1 Sensors

Without sensors, a robot is just a machine. Robots need sensors to deduce what is happening in their world and to be able to react to changing situations. This section introduces a variety of robotic sensors, explaining electrical use and practical application. Please do not be limited by the ideas contained in this section! The sensor applications presented here are not meant to be exhaustive, but merely to suggest some of the possibilities. Assembly instructions.

A standard plug configuration has been developed to connect sensors to the robot board (the brain) board, as shown in Figure 3.23 Notice that one pin is removed from the plug, making it asymmetric and therefore polarized. This means that once the plug is wired correctly, it cannot be inserted into a sensor port backwards. This makes the plug much easier to use. The sensor is connected to the plug with three wires. Two of the wires are used to supply power from the battery or any power supply to the sensor. These are the wires labeled \+5v" and \Gnd." The third wire, labeled \Signal" is the voltage output of the sensor. It's the job of the sensor to use the power wires (if necessary) and return its \answer", as a voltage.

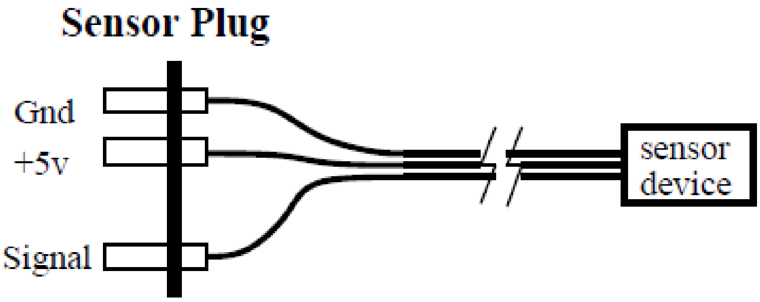


Figure 3.23: wiring a general sensor

3.7.1.1 Touch sensor

The most primitive, but often very useful, sensor is a touch switch. It is simply a pushbutton or other momentary switch that is mounted on a robot so that when the robot runs into something, the switch is triggered. The robot can detect that it has made contact with some object. Touch sensors used as collision detectors are imperfect in that it is hard to design a touch switch and bumper mechanism that can detect contact of any object from any angle. Creative mechanical design in implementing the bumper mechanism is important.

3.7.1.2 Bumpers

A bumper is a device that notifies your robot that you ran into an obstacle. When struck, the bumper moves and presses or releases a touch sensor, notifying your robot of the impact. Bumpers are the most common use for a touch sensor. This may be due to the Constructor using touch sensors primarily in this role, or it could be because bumpers are such useful devices.

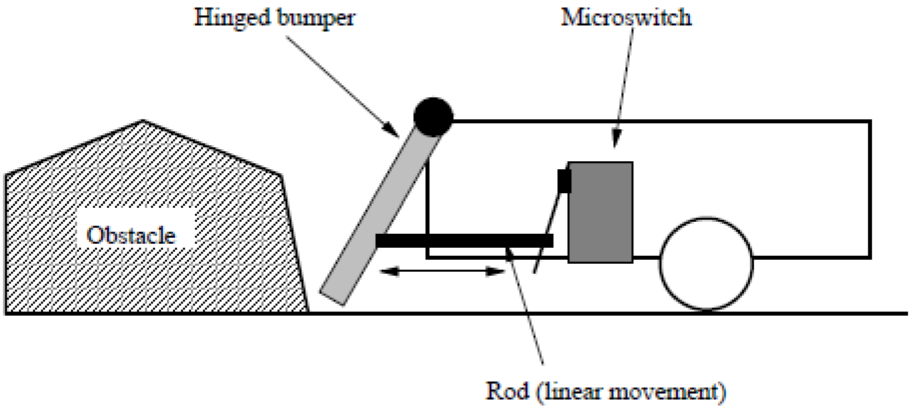


Figure 3.24: bumpers coupled to touch sensor

A bumper which shown in figure below assembly usually consists of four parts: the bumper, a sensor, a return mechanism, and a supporting framework. The bumper component is the part that receives the impact and converts it to motion that can be detected by the sensor. The sensor component is usually a touch sensor, but bumpers can be built using rotation sensors or light sensors

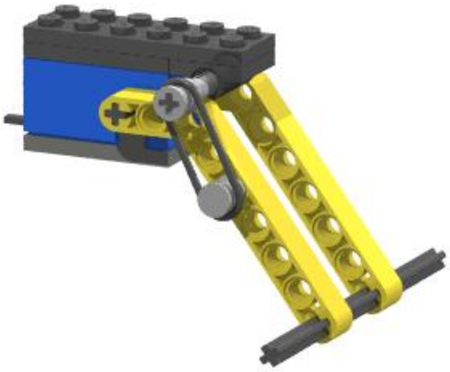


Figure 3.25: bumper uses rotation sensor

3.7.1.3 Light sensor

The light sensor contains a red light emitting diode (LED) and a photo transistor. The red LED illuminates the area in front, and the photo transistor measures the intensity of the reflected light. The light sensor returns a value in the range of 0 to 100%. The lowest reading ever seen is 20%, which could be taken at midnight with the lights off. You can get to 100% by aiming the sensor at the sun on a clear day or by holding it within a few inches of a 100W light bulb. In normal operation, the sensor values tend to be between 30 and 60%.

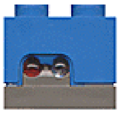


Figure 3.26 light sensor

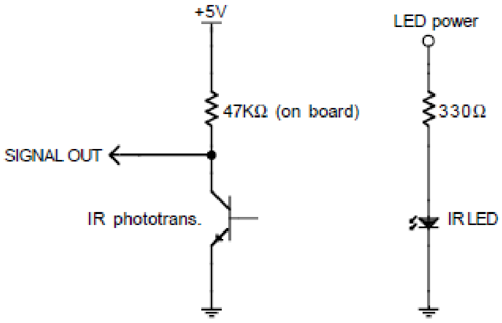


Figure 3.27: Light sensor with infrared emitter and photo transistor

3.7.1.4 The Robot Encoders

The encoder is a sensor attached to a rotating object (such as a wheel or motor) to measure rotation. By measuring rotation your robot can do things such as determine displacement, velocity, acceleration, or the angle of a rotating sensor. A typical encoder uses optical sensor(s), a moving mechanical component, and a special reflector to provide a series of electrical pulses to your microcontroller. These pulses can be used as part of a PID feedback control system to determine translation distance, rotational velocity, and/or angle of a moving robot or robot part.

For instance, if you have a wheel rotating, and you want to measure the time it takes to rotate exactly 40 degrees, or know when you have traveled X distance, you would use an encoder. The sensor would be fixed on your robot, and the mechanical part (the encoder wheel) would rotate with the wheel. The output of an encoder would be a square wave, so if you hook up this signal to a digital counter or microcontroller you can then count the pulses. Knowing the distance/angle between each pulse, and the time from start to finish, you can easily determine position or angle or velocity or whatever. Encoders are necessary for making robot arms, and very useful for acceleration control of heavier robots. They are also commonly used for **maze navigation**.

Calculating the robot motion with an encoder:

To do this calculation you need more information, such as **wheel diameter** and **encoder resolution** (number of clicks per 360 degrees, or counts per revolution). Starting off, you should know two things - wheel circumference and counts per revolution. Dividing the two, you can easily figure out the distance your robot travels between each encoder click:

Wheel circumference / counts per revolution = distance traveled per encoder count. Eq3.13

Now velocity is just distance divided by time . . . So using the answer in the above equation, divide that by the time passed determined from your microcontroller timer:

Distance traveled per encoder count / time = velocity. Eq3.14

After you know distance and velocity, you must then run a PID feedback control algorithm so that your robot can match a desired (pre-determined) distance and velocity.

Figure 3.28: Rotary Encoder

The Encoder can measure the distance or speed in both directions as we discussed in **Section2.4.** The Figure 3.29 below shows the Encoder simulation on MATLAB using the XPC target tools in addition to real time tools, while the two other figures are the results

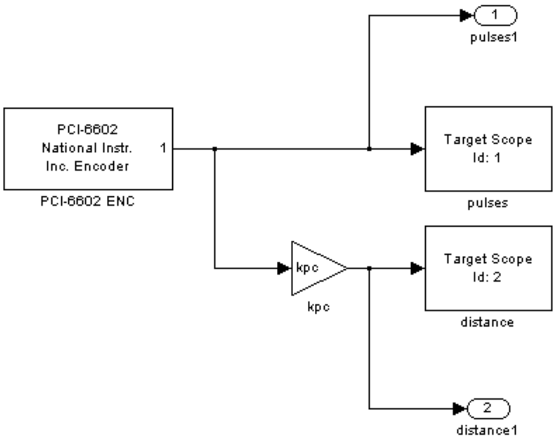


Figure 3.29: Encoder simulation on MATLAB

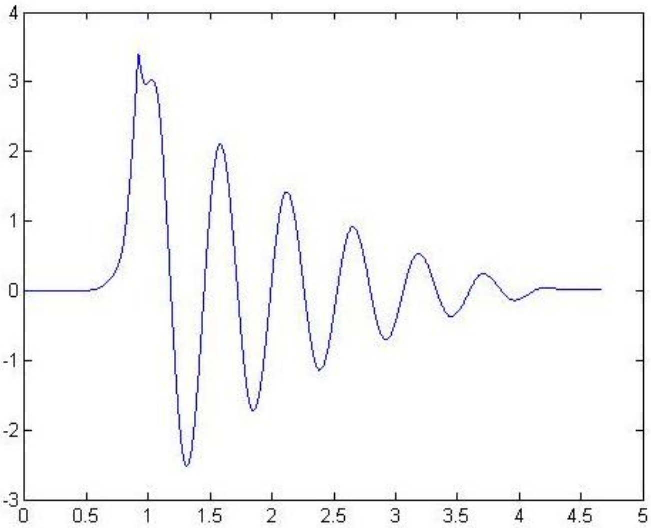


Figure 3.30: distance vs. time

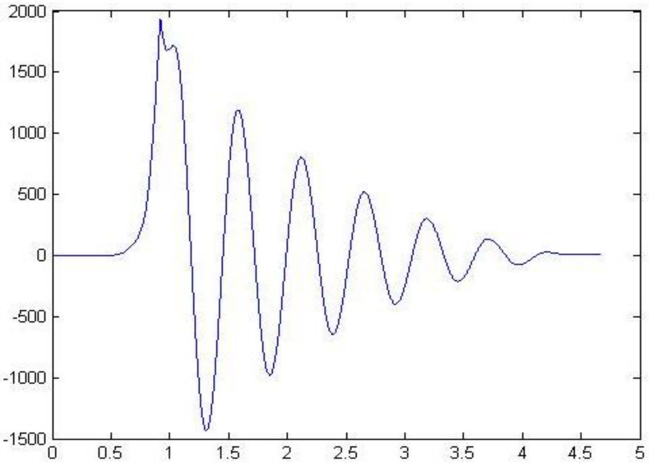


Figure 3.31: pulses vs. time

3.7.1.5 Ultrasonic sensor

This kind of sensor used to measure the distance from the robot by sending ultrasound wave, these waves couldn't been heard by human being according to range that we could hear between 20HZ to 22KHZ, after sending the waves from the transmitter of the Ultrasonic sensor they will be back to the receiver if there is an obstacle in front if not no waves come back and the robot keep moving forward, this sensor work on 5volt power supply and according to this equation we can measure the distance, in addition to the MATLAB simulation using XPC target tools.

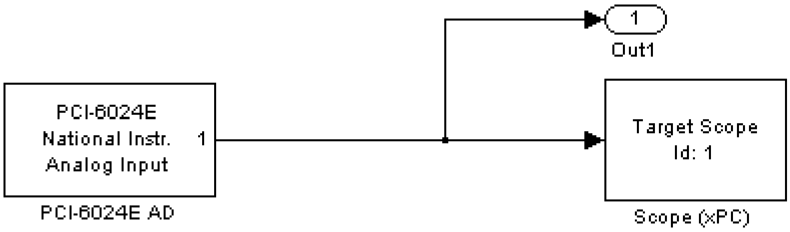


Figure 3.32: Ultrasonic sensor simulation

Speed = 2*Distance / Time. (3.15)

Distance = Speed * Time / 2. (3.16)

3.7.2 Motors and controlling modes

In this section we try to talking about the motors which we will use in our robot design, we will talk about the DC permanent servo motors with controlling encoder.

Servo are intended to be used in close loop speed and positional control systems where performance requirements are such that cannot be achieved by normal dc motor, the servo are designed to achieve good dynamic performance and steady state accuracy, its designed to achieve the same performance in both direction, high torque to inertia ratio, low friction and smooth ripple, free torque, in some servo, inertia is reduced by reducing the diameter and increasing length for the same rating. Small servos are usually with permanent magnet type. In this project we will use two small servos for two steering wheels.

As we mentioned formerly in the second chapter about the DC servo motor, we here try to clear the idea obviously, the figure 2.1.1 in the second chapter explain obviously the construction of the motor.

$$u = E + R_a.i + L_a.(dI_a / dt) \tag{3.1}$$

$$T_e = K_e.i \tag{3.2}$$

$$j = j_m + j_l \tag{3.3}$$

$$T_f = \sin(\omega).T_f + K_f.\omega \tag{3.4}$$

$$T_e = T_f + T_l + j.(d\omega / dt) \tag{3.5}$$

Where

T_e : *The electrical torque*

T_f : *The friction torque*

T_l : *Resistant torque*

j_m : *The motor inertia*

K_e : *Torque constant*

K_f: *The viscous friction*

j_l : *Load inertia*

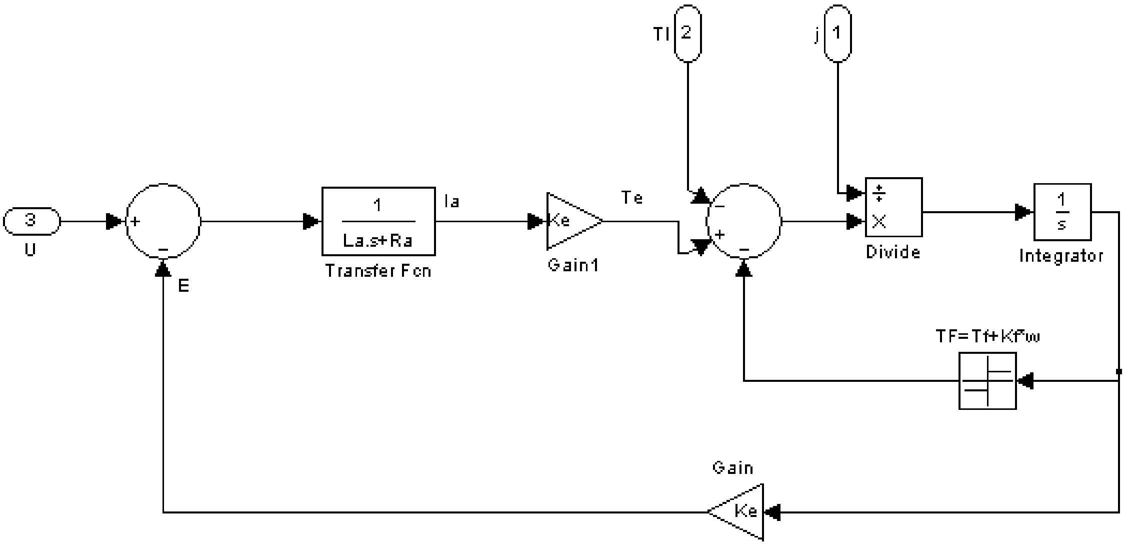


Figure 3.33: electric circuit of the armature

3.7.2.1 DC-Servo permanent magnet Motor simulation

Here we try to simulate the DC permanent magnet motor with driving mechanic load which we used in our project using the mat lab, referring to the figure 3.7.2.1 and using the equation below, and referring to the eq.(2.1)

By the simulation shown in this figure we connect the constant blocks to the inputs *U (10 volts), j(0.05 Kg.*m^2*) and* T_l*(4 N.m),* the internal parameters R_a*(0.5 ohm),* L_a*(10 mH)* K_e*(0.5 N.m/A)* and K_f*(0.001 N.m.s).*

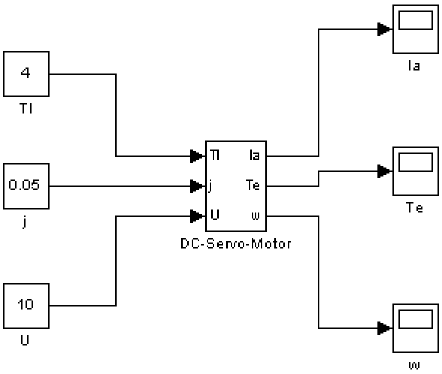


Figure 3.34: the permanent magnet DC-Servo motor simulation on MATLAB

The model after using the PID controller shown in figure below

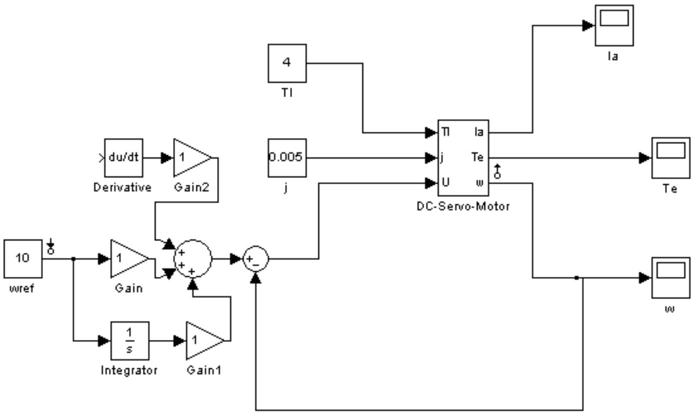


Figure 3.35: the model using PID controller

3.7.2.2 Speed control

In controlling the DC-motors there are three common methods, The **External resistance** which is useless in controlling robot motor because it has power dissipation with affect on Motor current, the second way is **Armature voltage** with is useless here. The speed control of our robot is depending on usage the H-bridge Technique, because it's useful in our application.

The NXT has three modes of controlling the motor--Off, On, and Float. In Float mode, the motor is not driven and can spin freely. It behaves just as it would if the connecting leads were removed. In Off mode, the NXT internally shorts the contacts of the output connector. This has a braking effect on the attached motor causing it to quickly stop spinning. In On mode, the NXT sends the motor either +9V or -9V to make the motor spin clockwise or counter clockwise. Pulse width modulation (PWM) is used to provide eight power level settings.

Pulse width modulation (as shown in **Figure3.30)** is an inexpensive way to control power output. Instead of setting the output voltage (voltage regulators are expensive), the NXT quickly switches the power on and off. Different power levels are achieved by varying the percentage of time that the power is on (this is called the duty cycle). On the NXT, the minimum power setting is zero, which corresponds to a duty cycle of 12.5% (the power is on 12.5% of the time). At level 7, power is supplied continuously (100% duty cycle).

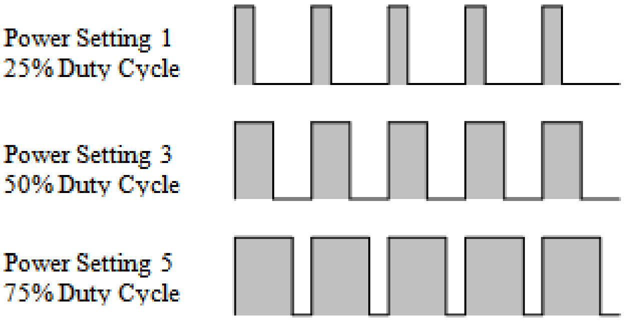


Figure 3.36: PWM duty cycle

The figure below shows the speed characteristic with time and step input after Linearize the model, the next figure also show the speed response to time with j=0.005.

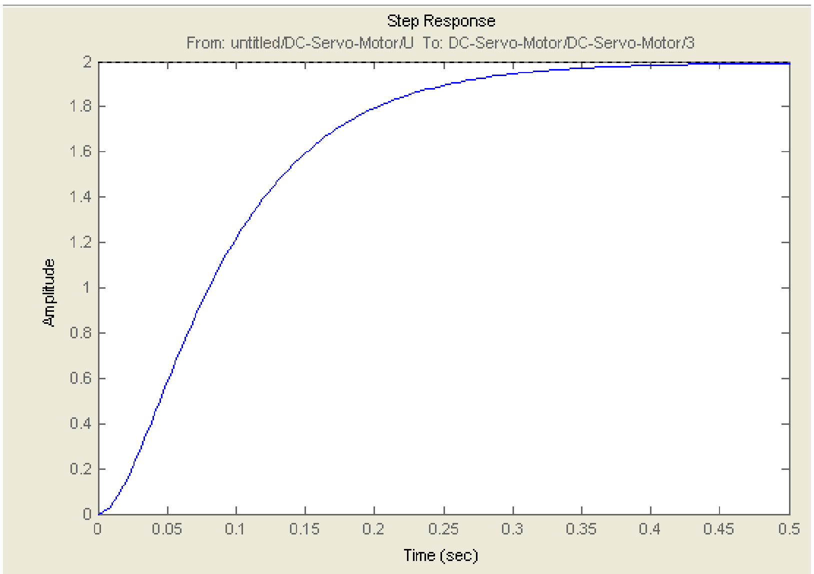


Figure 3.37: speed response versus the time at j=0.05.

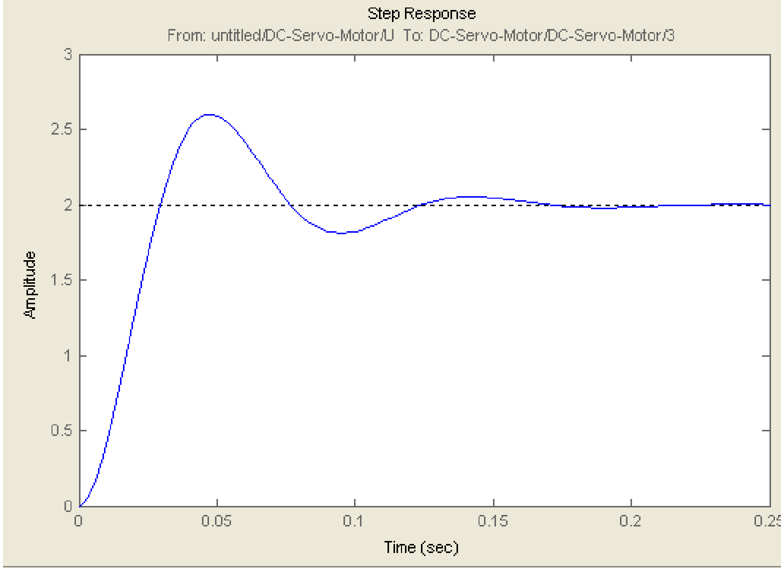


Figure 3.38: speed vs. time at j=0.005.

The figure below shows the connection of the DC-Servo motor on the H-bridge and simulates the system on the MATLAB, using the PWM and the PID controller.

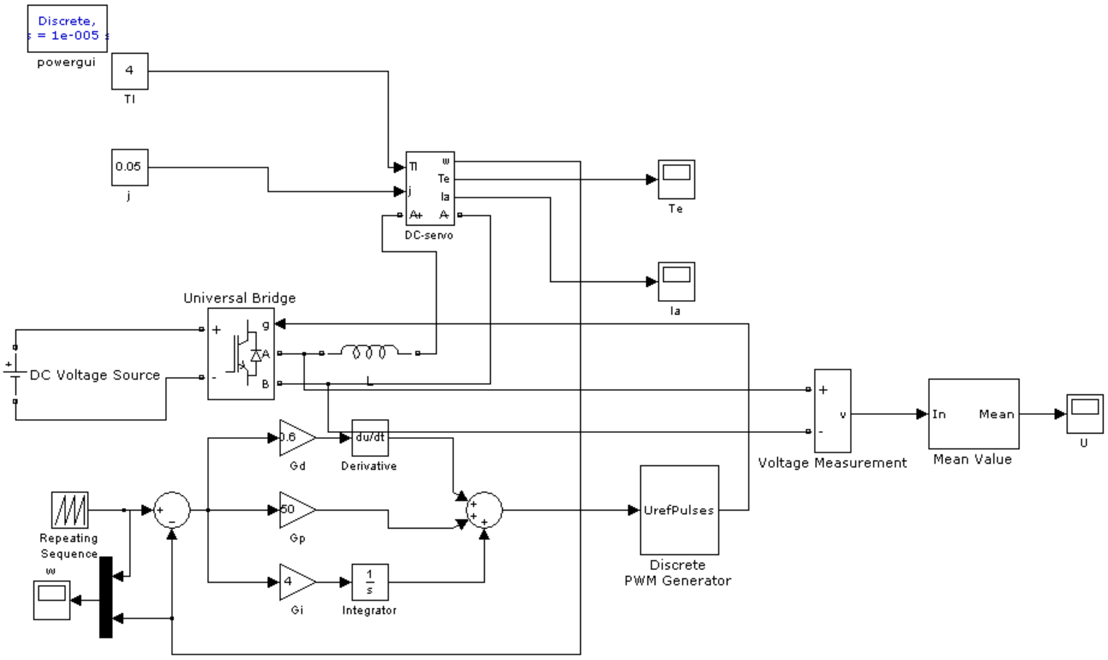


Figure 3.39: System Simulation on MATLAB

3.7.2.3 Motor Starting

Maximum current that dc motor can safely carry during starting is limited by the maximum current that can be commutated without sparking, for normally designed machines, twice the rated current can be allowed to flow and for specially designed machines it can be 3.5 times.

If a dc motor is started with full supply voltage across its terminals, a very high current will flow, which may damage the motor due to heavy sparking at the commutator and heating of winding. There for it's necessary to limit the current to a safe value during starting. When motor speed controlled by armature voltage control as mentioned before the controller which controls the speed can also be used for limiting the motor current during starting. In absence of such controller, a variable resistance is used by connecting it in series with the armature resistance and by changing the value of the resistance we can make a soft start; **Figure 3.31** shows the current behavior after simulation on MATLAB.

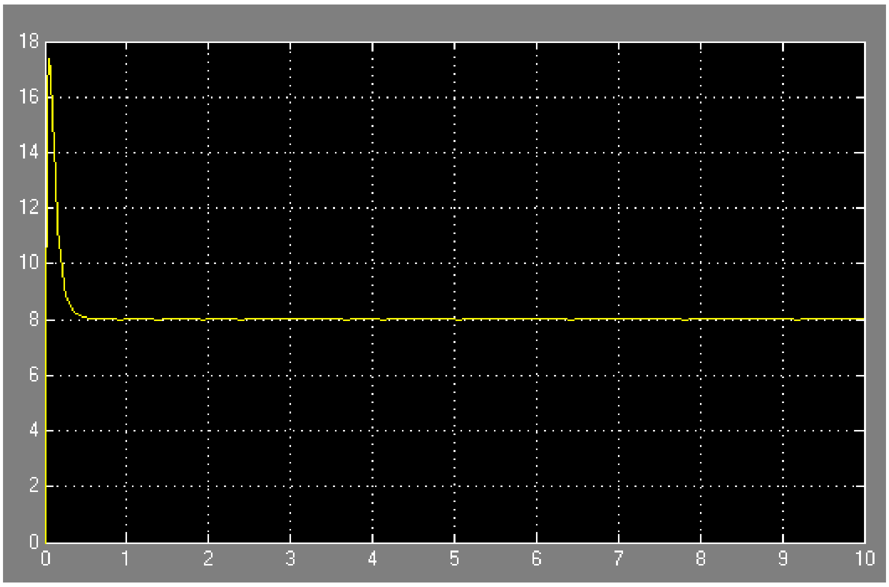


Figure 3.40: current behavior of the motor.

3.7.2.4 DC-Motor Braking

In braking the motor works as a generator developing a negative torque which opposes the motion. It is of three types, each way has its own benefits and draw backs, so I will go over each for you to decide which is best.

3.7.2.4.1 Controls Method:

This method requires an encoder placed onto a rotating part of your DC motor. You will have to write an algorithm that determines the current velocity of your motor, and sends a reverse command to your H-bridge until the final velocity equals zero. This method can let your robot balance motionless on a steep hill just by applying a reverse current to your motors.

3.7.2.4.2 Mechanical Method:

The mechanical method is what is used on cars today. Basically you need something with very high **friction** and wear resistance, and then push it as strongly as possible to your wheel or axle. A servo actuated brake works well.

3.7.2.4.3 Electronic Method:

This method is probably the least reliable, but the easiest to implement. The basic concept of this is that if you short the power and ground leads of your motor, the inductance created by your motor in one direction will power your motor in the opposite direction. Although your motor will still rotate, it will greatly resist the rotation. No controls or sensors or any circuits overheating. The disadvantage is that the effect of braking is determined by the motor you are using. Some motors brake better than others.

3.7.3 Motor source

The source of our robot motor is a rechargeable battery, 6 AA batteries these battery delivered DC voltage (9V), Robots may be powered by a variety of methods. For a small robot, however, battery power delivers a number of advantages over any other method. Batteries are cheap, relatively safe, small, and easy to use. Also, there are many different types of batteries, each with its own tradeoffs.

3.7.4 Robot electronics

The NXT is our robot brain, a tiny computer shaped, at the core of the NXT is a microcontroller running at 5 to 20MHZ with 32K of RAM. This may seem anemic when compared to modern desktop computers with 2GHz 32 bit processors and 512 Mbytes of memory, but it’s as powerful as the Apple II computer I learned to program on and significantly more powerful than the computers that sent men to the moon.

The microcontroller is used to control three voltage outputs, three sensor inputs, and an infrared serial communications port. Snap-on wire leads are used to connect the NXT to motors, lamps, touch sensors, light sensors, rotation sensors, etc… The serial communication port is used to download programs from a PC and can be used to communicate between two NXT’s.

The NXT is powered by batteries. Older versions of the NXT also had a 9V DC input plug for use with an optional AC adapter. If you have this model of NXT, the transformer will pay for itself in saved battery costs. Having two or three sets of good quality rechargeable batteries is also a good idea. The rechargeable alkaline batteries seem to work best with the NXT.

3.7.4.1 Programming

The NXT is programmed using special software that resides on your PC. The program is translated into byte codes and transferred to the NXT via an infrared serial link (using the IR tower). This method of programming--writing code on one type of computer to be run on another type of computer--is called cross development. A computer without a keyboard and monitor, like the NXT, is called an embedded system. Programs are written by picking instruction blocks and snapping them together.

Figure 3.41: The NXT programmable kit

3.8 Table design

The table with the robot will be set on it is made of wood with the proper dimensions(2.5m X 1.20m), in addition to that the lines painted on the table which considered as the robot guider such as the point that the robot will initialize and the homing the site of the obstacle, …etc.

3.9 Conclusion

The mobile robot is useful everywhere, they are so important to diminish the fault of position accuracy, so that in our robot we use the servo motors which they have good performance in position control by closed loop system, and give you very good results in position accuracy, we should notice the gears meshing in order to avoid the error estimation of the speed, in addition to avoid the sensor errors.

This robot due to cost of building it, you can use it in many application in our practical life in addition to the industrial aspects but you need to improve the size in addition to high power level, this robot in the future could be able to instead of the laborers in the factory and do works the laborers themselves couldn’t do.

3.10 Recommendations

There are several problems with using encoders for robot position control. First, just because your wheel rotates does not mean your robot is moving. Ever driven a car in snow? Error can quickly build up. This is why it is not recommended to use encoders for position feedback of your robot (such as in mazes). Second, encoders have a finite accuracy.

There are several things you need to watch out for when designing encoders. First, keep ambient light out of your sensor, such as sunlight. If light shines in to your sensor, it could potentially read false clicks. High resolution encoders for velocity control can take a lot of computational system time, so it is better to use a **digital counter IC** to count encoder clicks than to have your microcontroller count clicks. Your controller can read the counter value serially when it pleases instead.

Instead of using the NXT intelligent brick, you can use another IC’s that do the job, you can use the PIC18F4550 Microcontroller with pins and using the datasheet you can the function of each pin, in addition to that you need H-bridge IC, and you need an Isolation IC to protect the PIC from being damaged, such as the Opt-isolator ILD74, but the disadvantage of these IC’s is so hard to be programmed, the stepper motor could be used instead of using the Servo.

In this project you can connect a wireless camera in order to take photos of anything you want so that you will use a cable to connect the receiver of the camera with the NXT brick, and program the robot to do the proper task.

In order to get more efficiency and to let this work to be used in the indusrial factories we should improve the actuating to deliver more power for the heavy loads, such as improve the manipulator to work on hydraulic power or in DC 100 volt motor

To make this Robot automatically without pressing on the switch button to work, you can cotrol him by the remote by sending a signal through the remote control to the Robot and the Robot response to the order, or you can use the Bluetooth Technique.

3.11 Additional results

There are many kinds of gears in the life, but we got enough to talk about the spur gears because used widely in decreasing and increasing the speed, the gears are so close to the electrical transformer principles.

While attaching the gears together we should avoid the backslash problem which results from poor meshing of the gears. Backslash produce discontinuity and affect on the mechanical component and doesn’t give the best accuracy for controlling so that to solve this problem we try to use big gears, as we mentioned formerly in the gears section.

In our robot we can use treads instead of using wheels, the treads are popular in robots, and they have good traction on the rough terrain, they can cross ravines that would stop the wheeled vehicle, but unfortunately treads have number of disadvantages, they have fairly poor traction on the smooth surfaces. Also there is a lot power loss due to continuous bending and straitening of the treads as they roll around the sprocket wheels, for these reasons mentioned we use the wheels which give the robot the good speed it needs on any surfaces.

In programming on the NXT kit it is turned on, the NXT is receptive to messages coming in on its infrared serial port. With all the activity and congestion at competitions, it is quite possible that another team could unintentionally overwrite the programs stored in your NXT. Care should be taken to block the IR port when not in use.

It appears that the light sensor we could use is very sensitive to ambient light, which is disconcerting because you seldom have control over the lighting. So be careful when construction the robot using light sensors.

References

1. Fred G.Martin, The 6.270 Builder's guide second edition,The Massachusetts Institute of Technology, December 1992.

2. Thomas R. Kurfess Ph.D., P.E, Robotics and automation handbook, Printed in the United States of America, CRC Press LLC, 2000 N.W. Corporate Blvd., Boca Raton, Florida.

3. Gordon mccomb and myke predko, Robot builder's bonanza, The McGraw-Hill Companies, united states of America. 2006.

4. Ben-Zion sandler, Robotics designing the mechanisms for automated machinery second edition, Ben-Gurion University of the Negev, Beersheva, Israel.

Hyper links

http://www.solarbotics.net/library/pdflib/pdf/motorbas.pdf

http://services.eng.uts.edu.au/cempe/subjects_JGZ/ems/ems_ch12_nt.pdf

http://ocw.mit.edu/courses/mechanical-engineering/2-12-introduction-to-robotics-fall-2005/lecture-notes/

http://mindstorms.lego.com/en-us/Products/default.aspx/