UNIVERSITY OF PIRAEUS - DEPARTMENT OF INFORMATICS

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ – ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ

MSc Advanced Informatic Systems

ΠΜΣ Προηγμένα Συστήματα Πληροφορικής

MSc Thesis

Μεταπτυχιακή Διατριβή

Thesis Title: Τίτλος Διατριβής:	**Tactile Sensor for Robotic Hand** Αισθητήρας αφής για ρομποτικό χέρι
Student's name-surname: Ονοματεπώνυμο φοιτητή:	**Chrysoula Lioliou** Χρυσούλα Λιόλιου
Father's name: Πατρώνυμο:	**Apostolos Liolios** Απόστολος Λιόλιος
Student's ID No: Αριθμός Μητρώου:	ΜΠΣΠ 15046
Supervisor: Επιβλέπων:	**Dimitrios Vergados** Δημήτριος Βέργαδος

April 2024 / **Απρίλιος 2024**

3-Member Examination Committee

Dimitrios Vergados	Christos Douligeris	Dionisios Sotiropoulos
Professor	Professor	Assistant Professor

Guidelines:

1. **Research the existing HW properties of Kinova robot and OptoForce tactile sensor**
2. **Propose and design the method of connecting tactile sensor to ROS computer via Kinova robot arm.**
3. **Implement both HW and SW part of the proposed connection, document the solution.**
4. **Depending on the time available, implement recognition of grasp typical situations in ROS.**
5. **Perform experiments, document them, make conclussions.**

Executive Summary

Robot is a machine. It imitates the human activities. The robots are functioned according to the programming which is embedded in it. In robots, the Robotic ARM is separately used for some particular applications. It helps the human in the hazardous place of work and also it is used for research purposes. The Robotic ARM is mainly used to grip the objects, sensing the objects by using a robotic hand and also it is used in medical applications for testing purposes. The main characteristic of ARM robot is, it can able to sense its surroundings with the help of sensors. In this project, the tactile sensors are used in the Robotic ARM of fingers to sense the touching behavior of objects. Then the information from the robot is displayed on the PC. For this, hardware and software implementations are designed in this project. In the hardware implementation, the robotics finger with E field sensor is designed. The PCB is designed between an ARM robot and the tactile sensor. Then the information's from the sensor i.e., sensing information's of objects need to transfer to the Personal computer. For this transfer of information, another PCB is designed. In software implementation a robot and the movement of the robot according to the given instructions are design. The software implementation is implemented by using python. The method for gripping of objects by the robots is explained in hardware implementation part of this project. Then the sample of sensors which is used to interface with the microcontroller of the robot is also described in this project.

Table of Contents

1. Introduction

In modern industrial environment the need of the robots are too high. The robots became an unavoidable member in the industries (Both service as well as production) All the complicated as well as fatigue works are done by the robot system. There are huge number of varieties are available in the robot system based on the application (Adamatzky, 2014) (Aggarwal & Kirchner, 2014). Also the need of robot and automation systems in domestic purposes also increases. The robot technologies are also increases with the increase of the need of the peoples. But some complications are always there in the implementation of robot system. To remove the technical complexity involved in the robot system there are many research works are carried out by the many peoples (Alfadhel & Kosel, 2015). The whole world struggles to improve the performance of the robot system for assisting the humans. In this research work the deep research was carried out to improve the robot arm performance by employing the touch sensing. Kinova robot ARM was the most widely used robot system (Alsina, Escudero, Margalef & Luengo, 2007). It has the wider range of applications such as research, domestic use, and laboratory usage etc. Here the touch sensing for the robot finger was developed. Kinova robot ARM was used for the experimental investigation. Kinova robot system was the most common research purpose robot (Asaga, Takemura, Maeno, Ban & Toriumi, 2013). In this robot system the opto force tactile sensor was employed for sensing the touching action carried by the robot finger. Kinova Robot has the three fingers. These three fingers are used for the griping the objects (Basics, 2018). The problem is the robot not programmed to handle only one type of material. In laboratory used as well as some other uses the robots needs to handle many type of objects. It may be strong metal components or very soft elements or some brittle materials. The problem is to set the different griping forces for the various elements. Each time that can't be set manually ("Breakthrough in programming with new ActiveRobot™", 2001). If the same force applied for all the elements there is a possibility of getting damage or slipping from the robot hand. For example consider the two objects one is the iron ball (Campusano & Fabry, 2017)

. It needs very high griping force. And the same robot was used for griping the glass object. Here the robot system needs lesser force than the previous case. If the gripping force deviated from the required force leads to damage the components. For that the feedback control system will be required for the robot to sense the nature of the object. Here the opto force tactile sensor was placed in the each robot fingers. The opto force pressure sensor was the common tactile sensor used for the robot sensing. It has 8 pins for input as well as output processing. But the sensor was not directly connected with the robot processing unit. Because the robot doesn't recognize the data send by those sensors (Calinon, 2009) (Clarkson, 2000). Also the robot processor doesn't has the required ports for connecting the cables from the three sensors. For that the additional processing required. The additional processing system can gives the output lines in minimum number of pins. The processing unit must capable to consolidate the results of the three sensors. For that there are two additional PCB boards are developed (Colloquium on "Robot Sensors", 2001). That may contain the microcontrollers, signal conditioners, as well as the powering circuits. In this project the ATmega microcontroller was taken for this work ("cplusplus.com - The C++ Resources Network", 2018). The Tactile sensors are needed to mount on the Kinova robot ARM and get the information of robot through the PC. For that, two PCB designing's are needed. One PCB needs to connect the sensor to the robot. Another PCB is connected between the robot and the computer. The sense of touch or tactile sensing is the process of determining the physical properties of the objects based on the piezoelectric effect. This tactile sensor helps in sensing the normal force, shearing force and slippage. The tactile sensors sense static and dynamic pressure in a hand with high accuracy ("DELMIA™ Products for Perfomance - Dassault Systèmes®", 2018) ("Design Your Own PCB", 2018). It is mainly used in the finger of the robot or gripper system which is used to manipulate the task. So PCB is designed between the robot and sensors. Another PCB is designed to know the function of the robot. For that one PCB is connected between PC and Robotic ARM. In this research work the kinovo robot arm was needed to connect with the three opto-force tactile sensor to sense the gripping objects presence as well as the type of the griping object. Based on that the gripping force was needs to adjust. Here the detailed procedures as well as the basic details for the implementation of the system were explained.

2. Objectives

In order to ensure the accomplishment of the project the following things are need to be done successfully. They all are known as the key objectives of the project. And they are listed below.

- Conduct the detailed research on the system. Based on this study the adequate knowledge about the hardware system of the kinovo robot system was needed to be gained. Also research about the Opto Force tactile sensor and gain the knowledge about the system.
- After gaining the adequate knowledge about the hardware as well as software components related to the project, the proposal design should be made for the analysis. The proposed PCB design must contains two circuits. Among this the first circuit needs to connect the three sensors as well as the robot arm. It must facilitate the direct plugging of the three sensor cables. The second circuit was developed for establish the connection among the robot system and PC.
- Then the final process was to implement the various hardware elements as well as software elements based on the proposed design.
- Carry out the analysis on the robot arm to identify the object type currently the robot arm grasped. Also optimize the programs to vary the required grasping force required for grasping the various forces for the different materials. Also carry out some experiments on the developed system to analyze the performance of the developed system.

3. Literature Review

3.1 Human tactile sensing

Human tactile sensing

The human feeling of touch has filled in as the principal wellspring of knowledge and motivation for the improvement of automated material detecting. The most recent ten years have seen incredible improvement in seeing a portion of the components hidden in human contact. There is a critical refinement between two distinct parts of contact detecting in people sensation detecting alludes to the view of appendage movement and powers with inside receptors, while cutaneous detecting is the impression of contact data with receptors in the skin. The sensation receptors incorporate muscle axles, which react to changes in muscle length, and ligament organs, which sense muscle pressure (FANG & DING, 2012) ("FORTRAN | computer language", 2018). There are additionally receptors in the joints which report joint points and powers, in spite of the fact that the exact part of these sensors in engine control is the subject of some debate. Almost the majority of the muscles which impel the fingers are situated in the lower arm, with muscle strain transmitted to the point of activity by ligaments going through the wrist ("Fritzing", 2018). Investigations of robot controller configuration propose that transmission flow, for example, rubbing, kickback, consistency, and inactivity bode well and control endpoint positions and powers in light of actuator flags alone (Gong, He, Yu & Zuo, 2017). This suggests sensation data from the muscles is lacking for good control of contact, especially for the littlest movements and lightest powers where transmission elements tend to veil the coveted flag (Hashim et al., 2017) (Hughes & Hughes, n.d.).

Human touch can be astoundingly touchy. For shape detecting, the capacity to determine two pointed indenters on the fingertip requires that the focuses be isolated by no less than 1 mm. People see a surface as finished instead of seeing every little surface element exclusively if the highlights are not exactly around 1 mm in degree. It is discovered that the base noticeable tallness of a static raised component on a smooth surface was 0.85 microns. For all estimations of human material execution, it is essential to consider that mechanoreceptor reactions are unequivocally nonlinear and time fluctuating (Inc., 2018) (Jefferis, 2007). As a rule, human detecting and engine control data transmissions are moderate in the examination with mechanical controllers. Nerve conduction speeds are generally not exactly around 60 m/s. Latencies are no less than 20-30ms for the speediest reflexes and any longer for different reflexes and intentional reactions (Jung, Lee, Park, Ko & Lim, 2015). People likewise make up for moderate reaction times by controlling the mechanical impedance of the fingers, hands, and arms. Impedance tweak may confine the requirement for criticism in light of the fact that the right impedance can inactively create the suitable reaction to unsettling influences ("Kineo CAM: teachingless programming", 2012). It appears that human dependence on feed forward control is because of the long defers characteristic in nerve conduction. This is not an issue in apply autonomy since a controller can react to detected data unmistakably rapidly.

Tactile Sensing devices with manipulator

The sensor that has gotten the most consideration is the material cluster, which imitates the disseminated tangible game plan of human skin. These sensors ordinarily comprise of individual weight touchy components organized in a rectangular cluster over the contact surface of the fingertip ("Kinova Robotics - Robotics Business Review", 2018). As items come into contact with the sensor, the removal or weight at every individual component is estimated, which gives information of the nearby surface shape and add the weight dissemination over the contact between the robot finger and the protest. Average material exhibit sensors have around 8x8 components on 2 to 3 mm focuses (Martinez-Hernandez, 2015). The speculations exhibited here condense the bits of knowledge from these sources and present our theories for encouraging techniques for coordinating material detecting with control for control. Another essential material sensor is the fingertip compel torque sensor. This is a multi-hub stack cell mounted simply behind the fingertip that measures up to three-power and three-torque segments. This sensor does not quantify the subtle elements of the dispersion of contact weight, yet just the net power and torque vectors because of the contact with the protest. On a basic level, any kind of multi-pivot stack cell could be utilized for controller constrain torque detecting. Essential data about the contact can be gotten from inside estimations in the robot hand instrument. Robot hand joint point sensors are utilized with the kinematic model of the robot structure to discover the areas and introductions of the robot fingertips in a typical edge of reference. Dynamic material sensors are important in complex dexterous control assignments. These sensors react to changes in the conditions of the contact, in similarity with the quick adjusting (FA) mechanoreceptors in the human hand.

3.2 HWproperties of Kinova robot

A robot is a machine, which is programmed to perform the specific applications. The robotics functions are based on the software which is implemented in the robot. The software is embedded in the robot. The robot is also controlled by the external control device. By remote controlling the work of the robot is controlled in one place. Robots play a major role in heavy working areas. The robot arms consist of six power cubes and have seven degrees of freedom which is designed by AMTEC. There are six main types in industrial robots. They are articulated, Cartesian, Cylindrical, Polar, SCARA and delta. Robots are used in the field of welding, material handling, cleaning and drilling purposes. The robotic arm is a type of mechanical arm. It is programmable to do the work. ARM robots are widely used in the research of laboratories (Me et al., 2018) (Montaño & Suárez, 2018). The robot uses the interactions of minimum force to explore the surface of an object without changing its physical properties or size or without causing damage. The robots rarely control the angle at which its tactile sensor makes contact with the object. The software implementation is done on robots to communicate the problems and to solve it easily. The software implementation is completed by finding the solutions to the problem, identifying the current problem and communicates with other people regarding this problem. Solution to a problem is finding by writing the program to solve the complex mathematical operation. These operations are done by the robot when we give some input to it.

In robotics, the manipulator is a device which is used to manipulate the materials without any contact. This is very helpful in manipulating radioactive and bio-hazardous materials by using robotic arms. In a simple Arduino based ARM robot consists of one Arduino board and four servo motors (Mytum-Smithson, 2007). The power supply to those motors is not supplied from the Arduino. The servo motor has a separate supply to work. The servo motors used are one horizontal servo, two vertical servos and one holder servo (Newman, n.d.) ("OC Robotics provides snake-arm robot demonstration for Sellafield Ltd", 2012). The servo motors are highly used in automation technology. It rotates the part of robot machine with high efficiency and great precision. These motors are mainly used in toys, home appliances, cars, robots. The servo motor used in the robotic arm is Direct Current (DC) motor. It is used to get the power from the battery and run at the high speed and at low torque. The servo motor is controlled by pulse width modulation techniques. ARM robot mimics the human motion. It is controlled by human and it is programmable. It is designed in auto desk inventor. The improvements are easily applied. The elbow of arm is design by considering the following factors. The factors are rotation of a motor, rotation of gears and movement of forearm up or down. Omni wheels are used in ARM robot. It rotates like normal wheels. The smaller wheels rolled perpendicular to the rotational axis. It has two degrees of freedom. The degrees of freedom mean the number of values which has the freedom to vary ("Off-line programming trims time and cost of routing application", 2004). A typical robotic arm is made up of seven metal segments which are joined by six joints. The robotic arm is used as agricultural monitoring device to test the soil. It has the potential to reduce the environmental effects in farming. The robotic ARM senses the movement of human arm by accelerometer. The accelerometer generates an analog signal. Flex sensors are used to sense the movement of fingers in ARM robot. It causes a change in resistance. An impedance follower is used to convert the resistance into the voltage. Then analog to digital converter (ADC) is used to convert the analog signals into digital signals. These digital signals are sent to the microcontroller (Owen-Hill, 2018) (PCB Design, 2018). Microcontroller differentiates all data in three axes such as x, y, and z-axis. These data are used to generate the Pulse width modulation signals to run the motors in the robot. The data from the flex sensor are used to drive the motor driver in the ARM robot. This motor driver drives the DC motor.

Three modules are used to control the gestures in robotic ARM. They are sensor module, logical module and execution module. In sensor module the gestures can be captured by the movements of human arm, then accelerometer and flex sensor senses the movement and then it converts to analog output voltage. Accelerometer generates analog voltage. This voltage is passed to the low pass filter ("Pressure profile systems enters OEM market with new line of digital tactile sensors: first capacitive tactile sensors with direct digital output", 2006). Low pass filter is used to pass the low ranges of frequency. For gripping mechanism, flex sensors are used in two fingers i.e. forefinger and thumb. The change in the resistance occurs in the sensors depend the amount of bent. It converts the bent to electrical value. If more bent occurs means high electrical value is obtained (Pugh, n.d.). Flex sensors consist of carbon resistive elements within a thin flexible substrate. Analysis, Calculate and making decision are takes place in logical module of ARM robot. The analog voltage is given to the analog to digital converter. ADC converts the analog signal to digital signal and given it to the microcontroller. It generates 8 bit data in three axis with different voltages. At first accelerometer is in sleep mode. It can be activated by sending the positive high signal to SL pin in accelerometer through the microcontroller. In execution model, the ARM robot collects the data and follows the instructions ("RoboMind.net - Introduction", 2018). The working of mechanical part in the robot is done in three steps. In the first step the motor driver receives the signal. In second step, motor driver drives the robot. In third step, the robotic ARM movement takes place in the robot. Motors in the robot are used for joint rotation. There are three robotic arm parts. They are Base, Elbow and Wrist. The base consists of servo motor. It allows for forward and backward movements of ARM robot ("Robot control and programming", 2006). The elbow servo motors are used for up and down movement of ARM. The grip consists of DC motor. It is used to grip the objects in the ARM robot. The application of ARM robots includes the space shuttle remote manipulator system have multi-degree of freedom robotic arms. It is used in medical science for soft tissue manipulation, needle insertion, suturing and cauterization.

3.3 HW properties of OptoForce tactile sensor

The sensor system is designed to the use of an autonomous system in the unstructured environment. Each sensor is based on transduction principle. Transduction means conversion of energy from one form to another form ("Robot Library | RoboDK", 2018). The tactile sensor is a device which measures the information arising from the physical interaction with the object. Some typical sensor operational data are ultrasonic, resistive effects, capacitive effects, piezoelectric effects, visible light imaging, photoelectric and infrared, mechanical switching, inductive effects, thermal effects and Hall effects. The major components of the tactile sensor system are touch surface, transduction medium, structure and control/interface. The human touch is classified into two types. They are the cutaneous sensation and kinesthetic sensation. Cutaneous sensations are the pain, temperature, pressure. Kinesthetic sensations are the movement of the body, the position of the body and its equilibrium (Romeo, Oddo, Carrozza, Guglielmelli & Zollo, 2017). In other words, cutaneous sensing is internal sensing and kinesthetic sensation is external sensing. The advantages of resistive sensing elements used in the sensor are simple construction, durability, and easy to read out. The disadvantages are nonlinearity, hysteresis and low sensitivity.

There are three types in tactile sensors. They are Force / Torque sensor, Dynamic sensor, and Thermal sensor. Force/ Torque sensor is used to control the force by giving information to the object. The skin sensor provides accurate measurements of force in a larger bandwidth. Dynamic sensors are small accelerometer sensors. It is used in the fingertip of a robot. Three-axis accelerometers are orthogonally mounted to measure the acceleration in all the directions. The second type of dynamic sensor is the stress rate sensor. The third type is the thermal sensor. The thermal sensor detects the thermal gradients in the skin (Roth, 2004). The thermal gradients are the temperature and heat conductivity of an object. In the tactile sensor, there are three major contact types for sensing the force or some physical phenomenon. They are collision contacts, task contacts, and control contacts. Three-dimensional tactile sensors are developed by using

Capacitive, Piezoelectric, and optional sensing elements. MEMS technology is used to fabricate these types of sensors. Tactile sensors are used in Robotics, Computer hardware, Security system, Touch screen in mobile phones and for computing purposes. The advantages of tactile sensors are it does not have any external circuit or cables. The three-axis optical tactile sensor is capable to determine the normal and shearing force to the hands of the robot. It uses an optical waveguide transduction method by using image processing techniques. This sensor senses the normal and shearing force simultaneously (Trujillo-León & Vidal-Verdú, 2014).

The three-axis tactile sensor has high potential when compared to two and one axes tactile sensor. It consists of array sensing elements to sense the force. These sensing elements are made from silicon rubber, a light source, an optical fiber scope, and Charge Coupled Device (CCD) camera. The sensing elements are arranged on hemispherical dome structure. When the light is emitted from the source is directed through the optical fibers to the hemispherical dome. The feelers get collapsed when an object contacts the feelers. At that time, the light is reflected out of the acrylic surface. The light is reflecting out because of rubber. It has the higher reflective index. The contact phenomena consist of bright spots. These bright spots are the image of feelers collapse. These images are retrieved by the optical fiberscope. The optical fiber scope is connected to the charge coupled device camera. The information from the camera is sent to the personal computer. PC is used to control the procedure of dividing, digital filtering, integrated grayscale value and displacement of the centroid. It can be controlled by auto analysis program in the image analysis software Cosmos 32. The tactile sensor application includes manipulation of force and kinematics, exploration of thermal properties, friction and hardness and response the detection and reaction from the external agents. The primary physical mechanisms employed in sensors are cost, range, accuracy, repeatability, power requirements, and output signal specification, processing requirements, sensitivity, reliability, weight, and size.

3.3 SW Properties for both Robot and Sensors

The ROS (robot operating system) is the important framework to function the robot and it is the most used framework in the community of robotics. the SAIL (Stanford artificial intelligence laboratory supports artificial intelligence robot project if it is an Stanford robot. the ROS allows every user to reuse the code and it also improves the code quality by testing the numbers of users and number of platforms used. many institutions which bases research using ROS for running the hardware requirements and for sharing their codes to increase the time and reduce the redundancy ("Ultra lightweight robotic arm", 2018).There are some sensors are used in robot for identify the elements and some of the inputs are taken by the robot through the sensors. The sensors are temperature sensors, light, encoders and numerous others, and afterward open their estimations to ROS to create mechanical applications. Many sensors and actuators utilized as a part robotics technology are upheld by ROS plugins. Few organizations advantage from ROS and open equipment to make less expensive and simpler to utilize sensors, as existing programming can be utilized for them using very less cost. It also uses robotic arm manipulators, Cartesian control software, and other embedded controllers.

There are lot software properties are there to control and configure and to do all the tasks according to the specification and requirement of a kinova robot. To work with each and every hardware properties and to control the hardware parts. There must be a software is needed that is used to control and work with the hardware properties to function the robot. the main software properties are ROS kinetic frame work, debugging and visualizing tool, programming language for robotics framework, software for connecting with sensors, simulator for creating the 3D model etc. together these software models the robotics functions are implemented in the robot. Other than the mentioned software properties the citrix environments are also used for the robotics framework to function with the dot net framework. using the citrix environment robotics can implemented and installed but needs some requirements like it must license with the orchestrator and also even the time releases the name of the machine must not be changed.

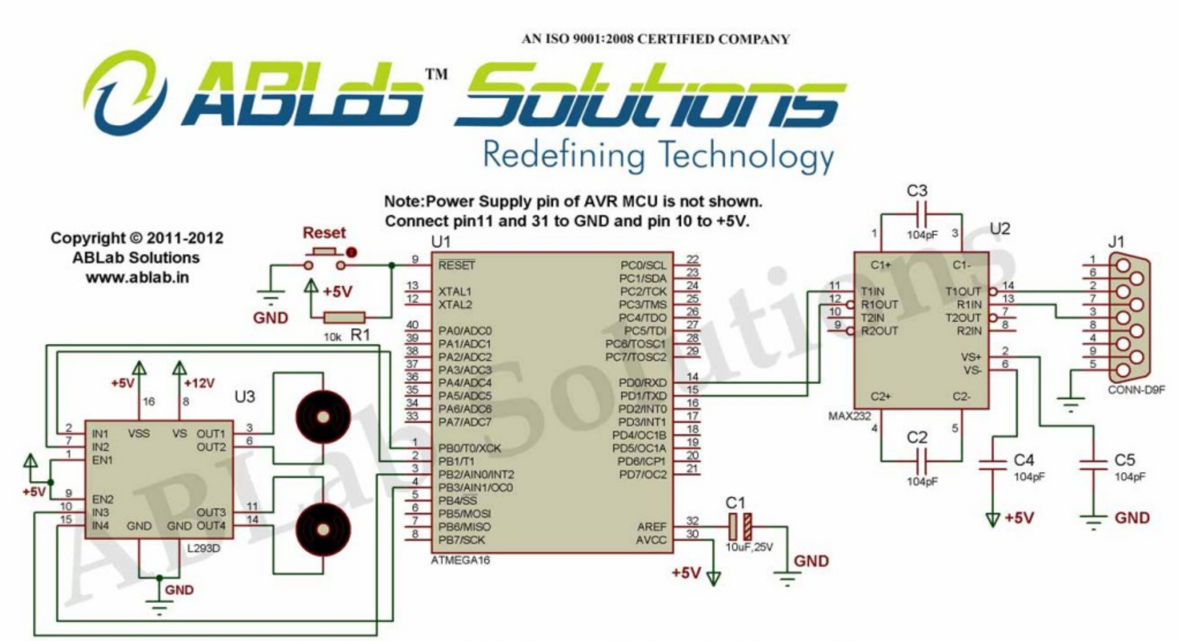
ROS gives an equipment reflection, initial level gadget control with ROS control, executions of generally utilized features and libraries, message going amongst procedures, and bundle administration etc. It utilizes chart design with a unified topology, where preparing happens in hubs that may get and send messages to speak with different hubs on the diagram net. A hub is any procedure that can read information from a sensor, control an actuator, or run abnormal state, complex automated or vision calculations for mapping or exploring in the robotics environment. The major software property for the robot is based on the ROS (Yuan, Dong & Adelson, 2017). It is the framework acts as a middleware and has collection frameworks for the development of robot using software's. Normally ROS is not OS but it provides device controls, computer clusters to function the hardware properties that is also functioned by the software properties. Creating knowledgeable robots that can achieve generation assignments not using the help men has entranced numerous AI consciousness networks. From a specialized perspective, independent robot route centers fundamentally around creating ideal worldwide ways to move the robot to an objective generation station in a continuous domain. Self-sufficient robots can responsively rectify their course by going around impediments with impact evasion and in the meantime investigate and outline unmapped area. At the point when the robot sees an objective from the separation, it may not be a precise separation. ROS is powerful tool or non-operating system in the evolution of robot operating system. Typically it has four distinct establishments, yet this relies upon the last utilize.

There are some minimum requirements are there for installing the Robot operating system. ROS uses the four installations steps but all the installation steps are based on the type of robot used. It differs for different kinds of robots according to the processor and hardware requirements. In the previous days the usage of robots are not The kernel of ROS gives naming and enrollment administrations to whatever remains of the hubs in the ROS framework. It identifies the distributers and supporters of points and in addition administrations. The part of the ace is to empower singular ROS hubs to find each other. Once these hubs have found each other, they speak with each other in a shared manner. It can found in a realistic case the means performed in ROS to promote a point, buy in to a theme, and distribute a message in the system to generate the output.

3.4 Connecting tactile sensor to ROS computer via Kinova robot arm

PC controlled Robot:

PC controlled Robot is designed with ATmega16 AVR Microcontroller. A microcontroller is a device used to control the operations in the embedded system. It is also referred to as System on Chip. It consists of memory and Input/output peripherals. AVR Microcontroller is developed by Atmel in 1996. It is Harvard Architecture of 8 bit RISC single-chip microcontrollers. ATmega 16 is an 8-bit Microcontroller. It consumes low power for its operations. ATmega 16 is based on enhanced Reduced Instruction Set Computer (RISC) architecture.

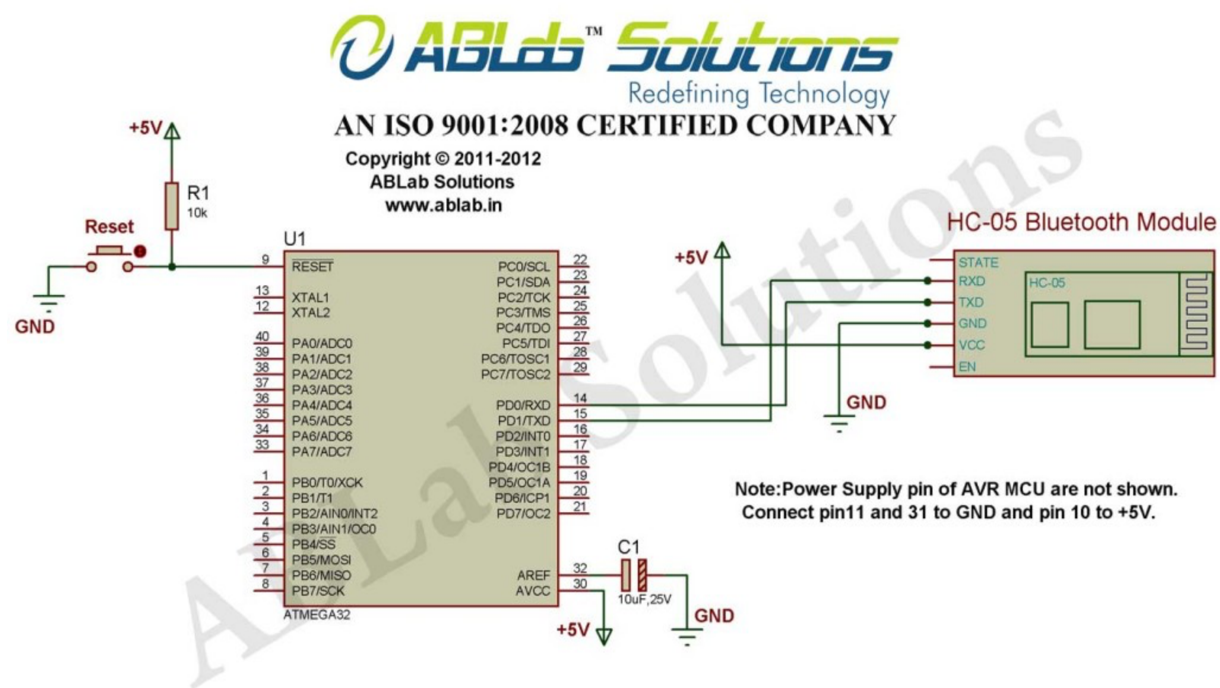


Circuit Diagram of PC Controlled Robot with ATmega16

The instructions for this microcontroller executes in one machine cycle. It has 16 KB programmable flash memory, Random Access Memory (RAM) is of 1KB and Electrical Erasable Programmable Read Only Memory (EEPROM) is of 512 bits. It is 14 pins Dual in-line package Integrated Circuit (IC). It has 32 Input/ Output lines which are divided into four ports. Each port size is 8 bits. The ports are named as Port A, B, C and D. ATmega 16 has inbuilt Universal Asynchronous Receiver Transmitter (UART), Analog to Digital Converter (ADC), Analog Comparator. In this controlling system, PC will send the signal directly to the Microcontroller through its serial port. The ASCII value is sent to the microcontroller from PC Keyboard. In New versions of PC or laptop, there is no such serial port. So it uses one USB. It connects to the serial converter cable in PC. The output from the PC is at RS-232 level. But in ATmega16 needs TTL/CMOS voltage levels. MAX 232 voltage converter is used to convert the RS – 232 voltages. This microcontroller reads its serial port output through USART (Universal Synchronous and Asynchronous Receiver/ Transmitter). If the signal receives from the PC means then the motion of the robot is controlled by using the motor in the robot. The robot also sends the signal to the PC about its position. The control keys of Robot are F or f – Forward, B or b – Backward, L or l – Leftward, R or r – Rightward, S or s – stop. The control keys are used to movement of the robot. This movement is possible by using servo motors.

Robot is controlled by using Bluetooth model:

Arunkumar Garg, Interface the Bluetooth to the Atmega 32 microcontroller. HC – 05 Bluetooth Module consists of 6 Pins. The power supply to the Bluetooth model is of 5V. The Receiver and Transmitter pin of the Bluetooth model is connected to the Receiver and Transmitted data of microcontroller. The power supply to the microcontroller is the same as that of 5V. HC-05 Bluetooth module is an easy to use Bluetooth Serial Port Protocol. It is designed for transparent wireless serial connection setup. This Bluetooth model is used in the master slave configuration.

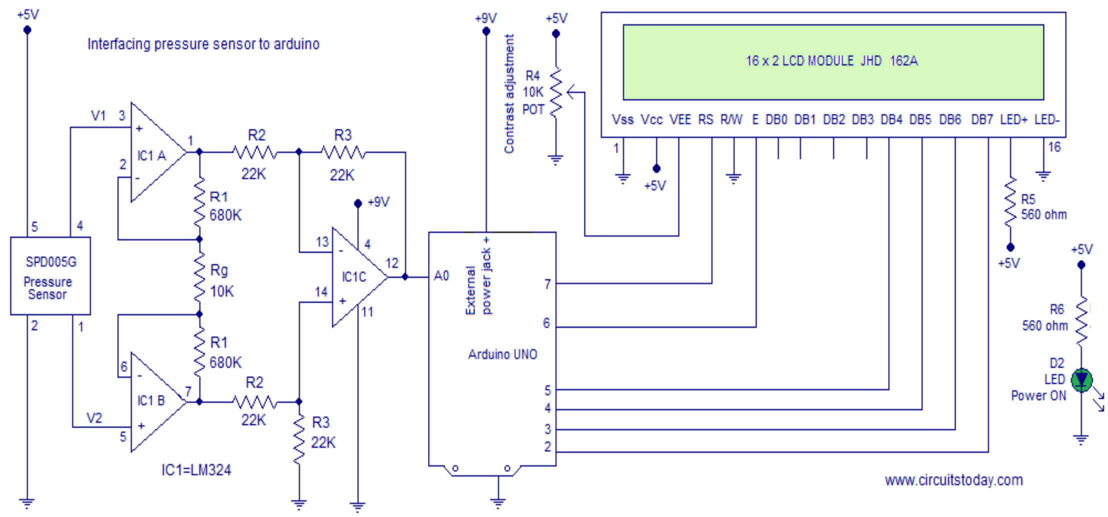


Circuit Diagram of HC-05 Bluetooth Module Interfacing with ATmega32

The communication between the Bluetooth model and Microcontroller is through the UART serial communication protocol. The RESET pin is connected to the 10KΩ resistor with the supply voltage of 5V. Reference capacitor of 10 μF is connected to the reference pin of the microcontroller.

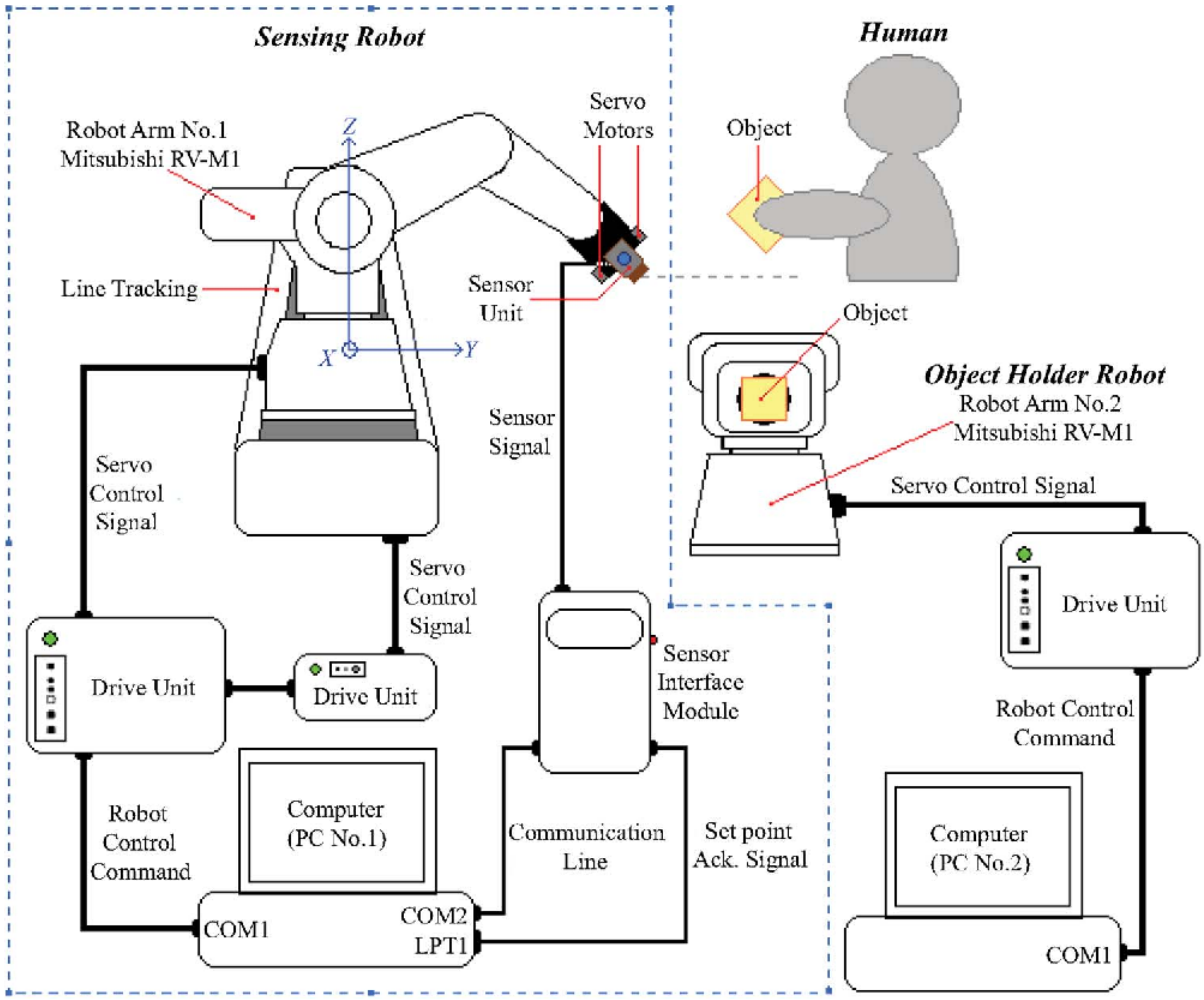
Interfacing sensor to the Arduino board:

In the interfacing of the sensor to the Arduino board, LCD display and sensor circuit board are used. Arduino board reads the inputs by sensing the light of sensor, a finger on the button, etc. The output from this board is blinking of LED, activating a motor according to the instructions given. Here, 16 X 2 LCD display is used to display the output from the sensor. Here the sensor which is used to feel the pressure. It consists of three operational amplifiers with resistors. The sensor circuit is connected to the voltage supply of 5V. The whole sensor circuit is connected to the A0 pin of a microcontroller. The pressure sensor is used to sense the pressure and send the signals through the three ICs which is below diagram. Then arduino UNO sends the signal information to the LCD to display the value of pressure.



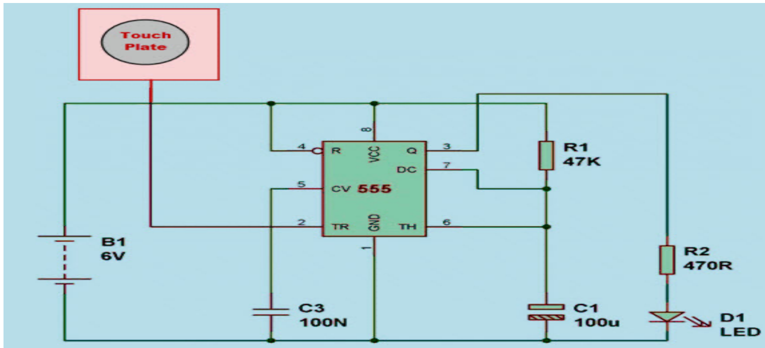
Block diagram of operation of ARM Robot with sensors:

According to Kitti Suvanratchatamanee, the sensing robot senses the objects in and around the environment. The ARM of a robot consists of sensors and motors. Motors are used to change the position of the ARM of robots. If the robot sense any object means first it sends the signal to the Sensor interface module. Then the acknowledgment and communication line is connected to the communication port of Personal Computer. Then from the robot control command is sent to the derived unit. The deriving unit derives the signal and sent as a servo control signal to the Kenova arm of the robot. The robot is moved in three axes as x, y, and z. These are the entire functions takes place in sensing robot. For example, if human paces an object in the object holder of robot means, then the sensor senses the object and sends the servo control signal to the drive unit. Then Robot control command is displayed on the computer through the communication port of the PC. Sensor interface model is connected between the fingers of the robot and the communication port of the computer. The servo control signal is given by the drive unit to the base of the ARM robot. The ARM of the robot are moving in three axis i.e., x, y and z. So it can able to move in all the directions.



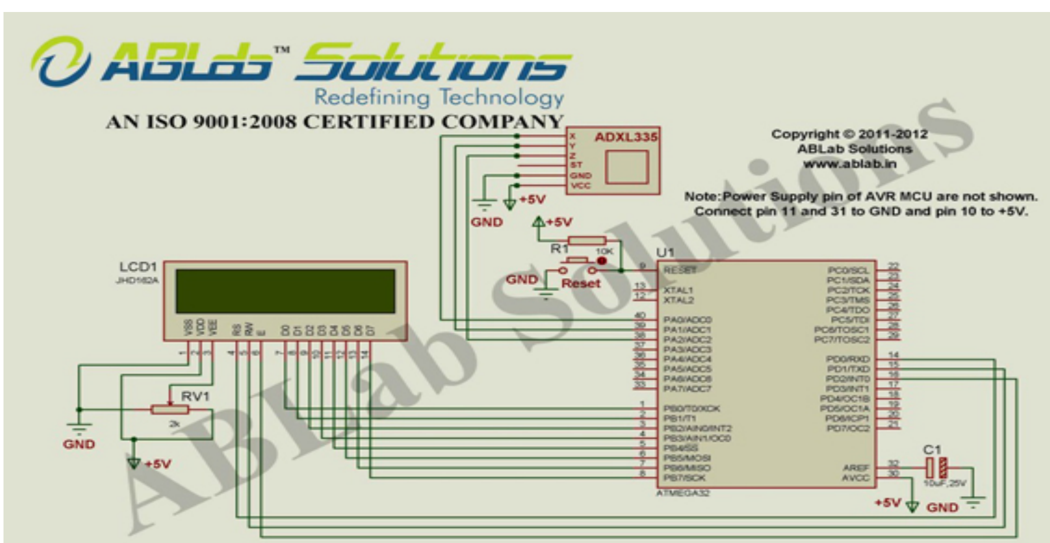
Tactile Sensor:

The sensor is used to sense the output by sensing the objects which are near to the robot. The sensor is classified into eight types. They are Temperature sensor, IR sensor, Ultrasonic sensor, Touch sensor, Proximity sensors, Pressure sensor, Level sensor, and Smoke and Gas sensors. In this, touch sensor senses the object by means of touch sensors. In the touch sensor, the IC555 timer is used with three resistors, two capacitors and one LED (Light Emitting Diode) with the battery voltage of 6 V. The trigger pin of the IC555 timer is connected to the touch plate. In the normal state, the touch plate is not touched means the LED is in off position. Once the touch plate is touched by anyone means, the LED is in ON position. The glow of LED shows the sensing of an object by using the sensor. The threshold and DC input of IC555 timer is connected inbetween a capacitor and the resistor. The third pin of IC555 timer is straightly connected to LED. But the resistor is used between the LED and Timer to control the flow of current.



Interfacing 3 axis accelerometer sensors to the microcontroller of robot:

ADXL335 three-axis accelerometer sensor is Micro Electrical Mechanical System (MEMS). It measures the acceleration resulting from both static and dynamic. Static acceleration is the gravitational force acceleration and Dynamic acceleration is vibration, force or motion. ADXL335 gives the three values in the axis of x, y and z. ATmega32 is connected with the sensor as well as LCD to display the sensing output. All the data inputs from the microcontroller is connected to the LCD data input of size 8 bits.. The power supply to the microcontroller, sensor and LCD is of 5V. Here the sensor is connected to the C port of microcontroller. The three-axis x, y, and z are connected to the first three data input of port C in the microcontroller. According to Arun Kumar Garg, if the object is at rest position near to the sensor means the LCD displays some value. Then the object is tilted to some other position means, the output value in the LCD is changing continuously



Circuit Diagram of 3-axis Accelerometer Sensor-ADXL335 Interfacing with ATmega32

4. New Design Proposal

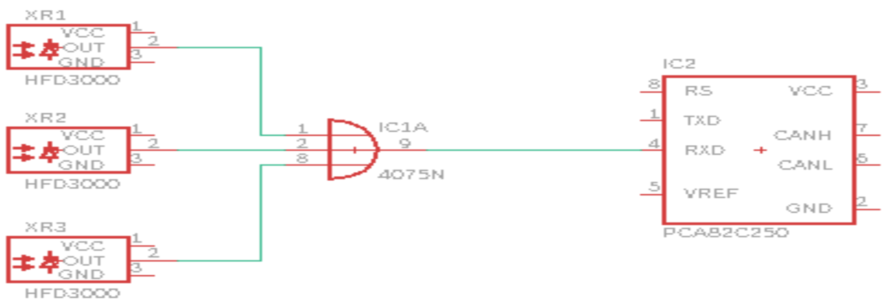
In this proposal, a new design is going to be implemented. Three tactile sensors are to be mounted on the finger of the robot. Then that sensor is providing information when grasping an object. This information needs to be displayed on the Personal computer. For this hardware and software implementations need to be created.

5. HW Design and Implementation

The below PCB design shows the E-field sensor in the finger of a robot. This design consists of a microcontroller, two transmitter channels, two receiver channels and some interfaces. The interfaces are USB through Future Technology Devices International (FTDI) chip and Inter-Integrated Circuit (I2C). Each transmitter and receiver pair has unique specifications and it measures the same measurement. Four different channel pairs are possible with the single finger of the robot. Left and right receive electrodes, mid and short range transmit electrodes are there in fingers. Each finger can also be connected to the third transmit electrode which is in the palm of the robotic hand. It provides additional left and right channels which works for a long range of values. The sensor boards are placed on the top of the transmit electrode board. The fingers also have the perpendicular receive electrode board on it. Two resonant transmitters are used to generate high voltage AC signals. Amplifiers are used to feed the signals to the microcontrollers ADC. Then enough processing takes place in the microcontroller to perform synchronous modulation on the received signal. Here 8-bit AVR microcontroller is used to control the flow of signals at 20 MHz. The transmission frequency is at the range of 156 KHz.

The grasping procedure is done by the combination of force and position control. Detection of the force of an object is needed for grasping process. The contact force threshold value should be higher to detect the light contact of the fingers with the object. The robotic fingers grasp the object based on the strain value collected at the target point of E-field in the finger of the robot. Mid-range and short-range sensors are used in the finger of the robot. The mid-range sensors are used for the preshaping controller with some points, it will keep the fingers of the robot a few distances away from the object. Once all the controllers are stabilized, then the preshape controllers moved to short-range sensors. IT will make the fingers of robots moved forward towards the object. Then the controllers are again stabilized to close the remaining distance of the object and grasp it. To record the encoder values at the point of contact with the object, strain gauges are used in the robotic fingers.

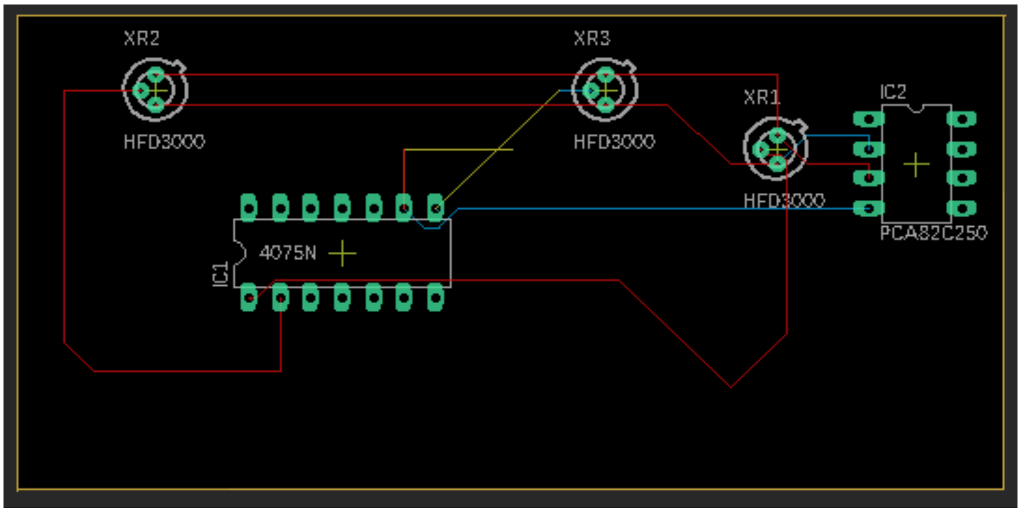
First PCB – Connection between sensor and Robot finger



The above schematic layout represents the first PCB design of this project. In this first , the sensors need too connecct with the Robotic ARM. So the sensors are need to be selected. This design is carried out in the Eagle editor. The sensors are selected from the eagle editor. In this sensors has three pins. One pin is for Vcc voltage connection to operate the sensor. The second pin is the Output pin. Third pin is the Ground pin. In this sensor, there is no input pin. Because it takes the input by sensing waves. It converts the analog sensing waves into digital waves to transmit the signals in digital form. This is the operation takes place in remaining three sensors and produce the digital output.

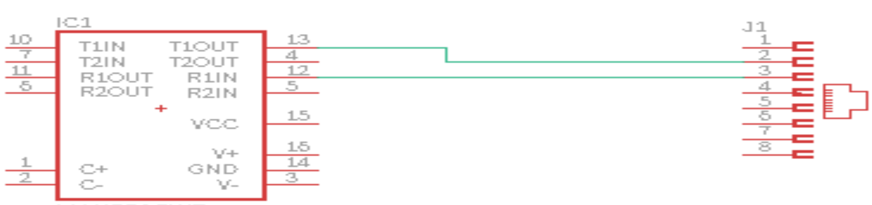
Now, the digital output from the sensors needs to send it to the robot. For this operation, the output from the sensors are gathered at one point. For this, OR gate is used. OR gate is a digital gate, it produces the result as high, when any one of the inputs is high.The aim is to transfer the information from sensors to robot. If any of the sensors are active means, then it sends the information to the robot. So OR gate is selected for this design. Here three input OR gate is used because of the three sensors.

The output from the OR gate is connected to the CAN connector (referred from the given attachments) as per the requirements mentioned in this project. The CAN connector is used to transfer the information between the microcontroller and other devices. Here other devices indicates sensors. This CAN connector can act as transceiver. Transceiver means transmitter and the receiver. In this design, it stores the data from the sensors and after getting full information from the sensors, it will send it to the robot.

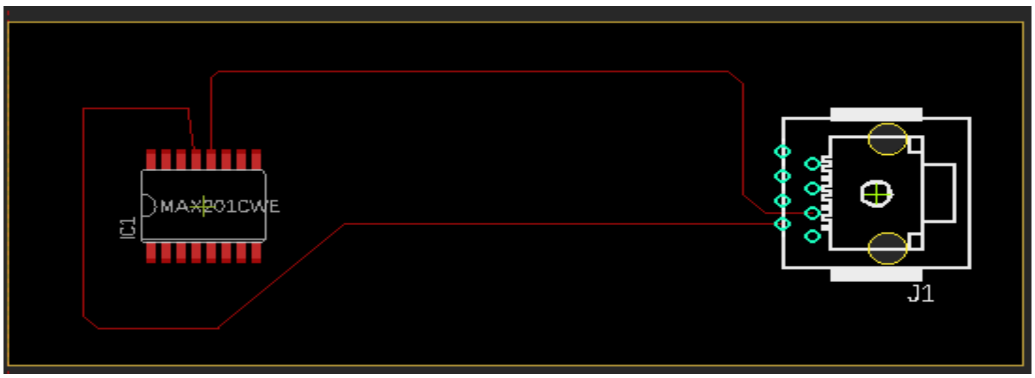


Here the ARM robot is considered as the microcontroller. Because microcontroller is used in all the robots to control their operations. The TXD pin of CAN connector is connected to MOSI (Master Output Slave Input) pin of microcontroller i.e., Robotic ARM. During this connection the CAN connector acts as the master and microcontroller acts as slave to get the signals from the master. So TXD pin is connected to the MOSI pin of microcontroller i.e. Robot. These are all the operations takes place in first PCB. The above diagram shows the PCB board layout of the first PCB design in this project.

Second PCB – Wired connection between PC and the Robot



The above PCB schematic layout shows the second PCB design. In second PCB, the robot needs to connect with the Personal Computer to transfer the robotic function information. As already descried in the first PCB design, here microcontroller is considered as the ARM robot. So the information output from the Robot is given to the PC by using RS232 IC. It was once used for connections to mice, modems, data storage, printers, uninterruptible power supplies, and other peripheral devices and has no multipoint capability, large voltage swing, large standard connectors and lower transmission speed. RS232 IC is used to improve the output voltage level according to the PC input voltage.

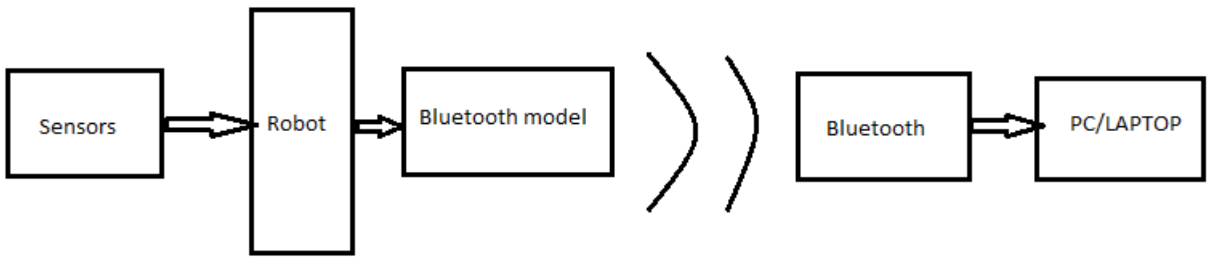


Since microcontroller produces the output at maximum of 5V. This 5V is not enough to connect the output signal to computer. Because, computer needs more voltage of input to it. So, RS232 IC is used to improve the voltage according to the PC. It is used to transfer the data for communication purposes. Here RS232 acts as the UART device. UART means Universal Asynchronous Receiver and Transmitter device. The UART devices are used to communicate the information between two IC's. Here IC RS232 is used as UART in this PCB design to transfer the information from the robot to PC. ("Sci-Hub: устраняя преграды на пути распространения знаний", 2018)

Second PCB – Wireless connection between PC and the Robot

Here microcontroller is considered as the robot. Because every robot has a microcontroller to control its all operation. So here microcontroller is considered as the robot. The information from the robot is transmitted from TXD pin of microcontroller to UART RX pin of the Bluetooth model.

The diagram is used for the wireless connection between the robotic ARM and the PC. This microcontroller Bluetooth model consists of UART-Universal Asynchronous Receiver Transmitter pin in it. It is used to interface the robot and the PC. To transfer the information, TXD pin of microcontroller is connected to UART RX pin in the Bluetooth model. The UART TX pin available in the Bluetooth model transmits the received signal from the robot to the PC which has in-built Bluetooth hardware. USB Bluetooth is required if not the PC or laptop has in-built hardware. Then the computer displays the information of the robot. This is the second PCB design. ("Interfacing Serial Bluetooth Modem with Computer using ATmega16 | EngineersGarage", 2018)



It shows the interface between the Robot Bluetooth model and the PC. HC-05 Bluetooth model can also be used for Bluetooth communication for the robot. Since it is the UART pin, it is also used to transfer information from computer to robot for controlling operations.

Another method of communicating the information from the robot to PC by Bluetooth is using Bluetooth modems. The Bluetooth modems are AUBTM, BLUESMIRF, RN41, HC-05 and HC-04. One of this modems are connected to the PC. Then in the PC under blluetooth iccon , the add device option needs to be selected. Then it displays the microcontroller i.e., robot Bluetooth module in the PC. Click on that option to connect with that particular device. After this step, system asks the pairing code to establish the connection between two devices. Then the pairing code need to be entered in that step. This pairing code is given in the datasheet of modem.

In this project, Connect the LCD module and Bluetooth modem to the Development Board. Note that TX pin of modem is connected to RX pin of the microcontroller and RX pin of modem to the TX pin of the controller. Now serial communication software need to communicate with the microcontroller and the PC. If the system has Windows XP software, then there is software called hyper terminal is there. Other than the XP like, windows 7, 8 and other versions. It needs to install one of the third party free software which is available on the net. These software is accessed by giving the information like Baud rate, port, data bits, parity and stop bits and hardware flow control. Then the LCD and USART implementation code needs to be created to transfer the information between the ARM robot and the PC. Here the LCD indicates the LCD in the microcontroller ("Sci-Hub: устраняя преграды на пути распространения знаний", 2018)

6. SW Part Implementation

According to the analysis transported out by the 'Aitken, J' on foundation of establish a rearrange manage device within the ROS for independent robots. ROS mention at Robot Operating System. The important parts of robots are developed based on ROS device. A proper model is described as a tripartite chart denoted the robots practical design of buildings. ROS devices are then generic to "System graph" is developed. Orthogonally of a source and device chart is described and repetition points of robot parts are examined for keep up fully worked of the robot automated reconstruction in meet to hardware breakdown. In this project discussed to allow AI analysis tools, referred as PDDL, to estimate legal reconstructions. I take this content of the project is a both of robotic arms what needs rearrange mentation of the based on control device offer to maintain the ability to perform to take out a problem. A machine (robot) is built in many number of metallic bones. The robotics necessary to few arrangement of well talented thinks to be approve to work its programmer, otherwise static statement or command by developing some at least fundamental, determination on its own. ROS is not an operating system, it supply work developed for miscellaneous system cluster referred as hardware ideas, poor level system management, execution of generally used capabilities, response between methods, and package management. Working sets of ROS-under methods are denoted in a chart architecture where processing alert location in growth that may collect, send and multifarious sensor, control, management and another information's.

The benefit of an OS for robots

A past years consider at before ROS, each humanoid developer and robotics investigator would invest worth able amounts of time discover the embedded application within a humanoid, include the hardware itself. This referred talents in engineering and fixed programming. Likely, the applications developer use this platform and support an embedded programming. The important aim of a humanoid OS is to discard repeated recreate the wheel, and to allocate static ability working hardware abstraction.

Organization of ROS

ROS's knowledge can be compile the following five main laws, are

Peer to peer, open source, multi-language, tools based, thin.

Peer to Peer: A capacity difficult machine consists of various suitable systems or boards joined way to Ethernet or off board systems for exhaustive calculation problems.

Open source: It runs any platform and it's free of cost. Is a big advantage of ROS.

Multi language: ROS is language-ideal, and can be developed in different languages. The ROS important service at the peer to peer links is discuss in XML-RPC, which still alive in a best types of languages.

Tools Based: ROS accept a microkernel draft, which makes a many list of tiny software to configure and implement the different ROS parts.

Programming with ROS

ROS is autonomous language. In this time, three important building functions have been declared for ROS, discovering it able to application ROS in programing languages. Likely python, Lisp or C++. Two extra practical building functions are referred, generating it able to application ROS in java or Lua.

ROS File system:

Two main principles are, package and stack.

Package: The basic group within ROS tool arrangement. A Package is a guide including joints, outside functions, information construction files and single XML arrangement files are placed.

Stack: Describes at group of packages. It refers a group of basics. They are navigation, locationing.

ROS stands for Robot operating system. ROS is the emerging one for controlling process. It contains the open source software that can be used for the purpose for creating the robots and also logic program controllers known as PLC. The software are flexible one and expandable one. They need the much better than the other tools. Robots are easier to develop the mobile application. The SwRI is the one of the best open source robotic software (Kumar, 2018).

Robotic Controls:

PLC is used for the purpose of machine function controls and also integrating. PLC stands for programmable logic controllers. It used for the machine control purpose.

Industrial integration of Robotics:

Software are very flexible one for creating the Robots. Most technology are used to develop the Robots controls. Software integrate the effective solution for robotics and also control function and interface. It makes the principles like create the design. The feature of the robots is seamlessly. The ability of creating the Robots system are most sensitive. The benefits of the robots are cost reduce, less wiring, share the platform easily (arm, 2018).

ROS tech developments:

The ontology driven IDE is the best one to create the Robots. The main purpose of the IDE to provide the graph and flow change of the exact system. It can further support the nodes for drop drone function. It's useful for creating the Robots app easy and also used to identify loosely. The new application named as Reapp. Reapp is mainly used to manufacturing the robots for the model based upon the ROS system.

The team SWRI creating the new Plugin named as QT. It provide the templates for the ROS. It very useful to developing the new one named as the graphical 3-D Tool. The most powerful tool easily to design and creating the Robots system. The graphical setup tool named as CAD – to-ROS. The 3-D tool are easily to create and modify.

RIC (ROS-Industrial Consortium):

All robot deserve with the variety of capabilities. The autonomous mobile robot can move space freely. This robot is controlled by itself. FTP winners presented five proposed FTP titles. They are ROS based graphical user interface, mobile manipulation demonstrator based on movie, deburring demonstrator based on movie, the force control system for grinding, growth to a human tracker system for safety (2018).

The general concepts of the ROS:

It’s the open source framework software and it could be working as the group.

The ROS is improve the flexible and efficiency of developing the technology. Thus the Robots to work safety. The python code :

```
from __future__ import print_function
importos
import sys
importsubprocess
import threading
importrospkg
importrosmake
```

```
## make sure that rospack is built, it is a requirement for rosmake
defassert_rospack_built():
    p = subprocess.Popen(["rospack","help"], stdout=subprocess.PIPE, stderr=subprocess.PIPE)
output = p.communicate()
ifp.returncode != 0:
print("rospack not available.\nPlease install rospack before using rosmake and make sure it is available on your path. %s %s"%(output[0], output[1]), file=sys.stderr)
sys.exit(1)
result = 1
withrosmake.Printer():
rma = rosmake.RosMakeAll()
try:
ifrma.main():
result = 0
exceptrospkg.ResourceNotFound as e:
print("cannot find required resource: %s"%(str(e)))
#make sure the thread is done
rosmake.Printer().join()
for t in threading.enumerate():
if t != threading.currentThread():
        # Join all threads before exiting
print("Cleaning up thread", t)
t.join()
sys.exit(result)
```

Setup file:

```
fromdistutils.coreimport setup
fromcatkin_pkg.python_setupimportgenerate_distutils_setup
d =generate_distutils_setup(
packages=['rosmake'],
package_dir={'': 'src'},
scripts=['scripts/rosmake'],
```

```
requires=['rospkg']
)
setup(**d)
```

Publisher and subscriber Node code for C++:

Node creation:

```
cmake_minimum_required(VERSION 2.8.3)
project(beginner_tutorials)

    ## Find catkin and any catkin packages
find_package(catkin REQUIRED COMPONENTS roscpprospystd_msgsgenmsg)

    ## Declare ROS messages and services
add_message_files(DIRECTORY msg FILES Num.msg)
add_service_files(DIRECTORY srv FILES AddTwoInts.srv)

   ## Generate added messages and services
generate_messages(DEPENDENCIES std_msgs)

  ## Declare a catkin package
catkin_package()
```

Make List coding:

```
add_executable(talker src/talker.cpp)
target_link_libraries(talker ${catkin_LIBRARIES})
add_dependencies(talker beginner_tutorials_generate_messages_cpp)

add_executable(listener src/listener.cpp)
target_link_libraries(listener ${catkin_LIBRARIES})
add_dependencies(listener beginner_tutorials_generate_messages_cpp)
```

Implementing the python code in ROS:

Turtlesim Namespace used to generate the code in python:

Step1: Creating the file and add the directory.

First create the python file named as mynewone.py file and also creating the packages to load the current directory. The below code are the package loaded.

$ chmod u+x ~/catkin_ws/src/turtlesim_cleaner/src/mynewone.py

Step2: Code Understanding:

First Creating the class named as TurtleBot. After creating the class follow the Subscriber. It's used to provide the services of the robots. Next find out the Euclidean distance. In this method used to save the positon of the turtleBot class. After finished the distance method further create the PID controller .The PID controller is mainly used to move the robots as the right way. It will be contains the Linear speed and angular velocity. The speed must access the distance between the goals of the robot point. Choosing the distance dependent on the x-axis and y-axis. The both x-axis and y-axis are constant one. The python code are given below.

```
import rospy

   from geometry_msgs.msg import Twist

   from turtlesim.msg import Pose

   from math import pow, atan2, sqrt

  class TurtleBot:

    def __init__(self):

        # Creates a node with name 'turtlebot_controller' and make sure it is a

        # unique node (using anonymous=True).

         rospy.init_node('turtlebot_controller', anonymous=True)
```

```
    # Publisher which will publish to the topic '/turtle1/cmd_vel'.
    self.velocity_publisher = rospy.Publisher('/turtle1/cmd_vel',
    Twist, queue_size=10
    # A subscriber to the topic '/turtle1/pose'. self.update_pose is called
# when a message of type Pose is received.
self.pose_subscriber = rospy.Subscriber('/turtle1/pose',
Pose, self.update_pose)
self.pose = Pose()
    self.rate = rospy.Rate(10)
def update_pose(self, data):
"""Callback function which is called when a new message of type Pose is
received by the subscriber."""
self.pose = data
self.pose.x = round(self.pose.x, 4)
self.pose.y = round(self.pose.y, 4)
def euclidean_distance(self, goal_pose):
"""Euclidean distance between current pose and the goal."""
return sqrt(pow((goal_pose.x - self.pose.x), 2) +
pow((goal_pose.y - self.pose.y), 2))
def linear_vel(self, goal_pose, constant=1.5):
    """See video: https://www.youtube.com/watch?v=Qh15Nol5htM."""
return constant * self.euclidean_distance(goal_pose)
def steering_angle(self, goal_pose):
```

```
"""See video: https://www.youtube.com/watch?v=Qh15Nol5htM."""
return atan2(goal_pose.y - self.pose.y, goal_pose.x - self.pose.x)


    def angular_vel(self, goal_pose, constant=6):
"""See video: https://www.youtube.com/watch?v=Qh15Nol5htM."""
return constant * (self.steering_angle(goal_pose) - self.pose.theta)
    def move2goal(self):
        """Moves the turtle to the goal."""
        goal_pose = Pose()
        # Get the input from the user.
        goal_pose.x = input("Set your x goal: ")
          goal_pose.y = input("Set your y goal: ")
      # Please, insert a number slightly greater than 0 (e.g. 0.01).
       distance_tolerance = input("Set your tolerance: ")
       vel_msg = Twist()
        while self.euclidean_distance(goal_pose) >= distance_tolerance:
            # Porportional controller.
             # https://en.wikipedia.org/wiki/Proportional_control
            # Linear velocity in the x-axis.
              vel_msg.linear.x = self.linear_vel(goal_pose)
              vel_msg.linear.y = 0
              vel_msg.linear.z = 0
             # Angular velocity in the z-axis.
```

```
            vel_msg.angular.x = 0
            vel_msg.angular.y = 0
            vel_msg.angular.z = self.angular_vel(goal_pose)
            # Publishing our vel_msg
            self.velocity_publisher.publish(vel_msg)
            # Publish at the desired rate.
            self.rate.sleep()
        # Stopping our robot after the movement is over.
        vel_msg.linear.x = 0
        vel_msg.angular.z = 0
        self.velocity_publisher.publish(vel_msg)
        # If we press control + C, the node will stop.
        rospy.spin()
if __name__ == '__main__':
    try:
        x = TurtleBot()
        x.move2goal()
    except rospy.ROSInterruptException:
        pass
```

Before run this code we are install the packages named as rospy and geometry_msgs. After install all the above packages using the math function, such as sqrt and atan. The publisher and subscriber message type contains the turtlesim.msg file. The below command are used to execute.

$ rostopic info /turtle1/pose

The below screenshot are following to run the command:

```
~$ rostopic info /turtle1/pose
Type: turtlesim/Pose

Publishers:
 * /turtlesim (http://birnuc1:45049/)

Subscribers: None
```

```
~$ rosmsg show turtlesim/Pose
float32 x
float32 y
float32 theta
float32 linear_velocity
float32 angular_velocity
```

The above screenshot are used to execute the x and y axis co-ordinates they are linear velocity, angular velocity.

Explain the python code step by step:

Step1: The above python code are explain in detail about the _int_ method function.The method function contains the object like publisher and subscriber.

```
def __init__(self):
    # Creates a node with name 'turtlebot_controller' and make sure it is a
    # unique node (using anonymous=True).
     rospy.init_node('turtlebot_controller', anonymous=True)
    # Publisher which will publish to the topic '/turtle1/cmd_vel'.
    self.velocity_publisher = rospy.Publisher('/turtle1/cmd_vel',
   Twist, queue_size=10
   # A subscriber to the topic '/turtle1/pose'. self.update_pose is called
# when a message of type Pose is received.
self.pose_subscriber = rospy.Subscriber('/turtle1/pose',
```

```
Pose, self.update_pose)
self.pose = Pose()
    self.rate = rospy.Rate(10)
```

Step2: Another one function is upadate_pose .This is the callback function it can be used for subscriber. The method pass the attributes named as self.pos.

```
  def __init__(self):
     # Creates a node with name 'turtlebot_controller' and make sure it is a
     # unique node (using anonymous=True).
      rospy.init_node('turtlebot_controller', anonymous=True)
     # Publisher which will publish to the topic '/turtle1/cmd_vel'.
     self.velocity_publisher = rospy.Publisher('/turtle1/cmd_vel',
    Twist, queue_size=10
    # A subscriber to the topic '/turtle1/pose'. self.update_pose is called
# when a message of type Pose is received.
self.pose_subscriber = rospy.Subscriber('/turtle1/pose',
Pose, self.update_pose)
self.pose = Pose()
    self.rate = rospy.Rate(10)
```

Step3: Find out the Euclidean distance method between the positions like that mynewone and turtle position.

```
def euclidean_distance(self, goal_pose):
"""Euclidean distance between current pose and the goal."""
return sqrt(pow((goal_pose.x - self.pose.x), 2) +
```

pow((goal_pose.y - self.pose.y), 2))

Step4: create the goal_pos function. The function mainly used to get the input values from the user. It can be used as the same datatype.

```
def move2goal(self):
  """Moves the turtle to the goal."""
  goal_pose = Pose()
  # Get the input from the user.
  goal_pose.x = input("Set your x goal: ")
    goal_pose.y = input("Set your y goal: ")
```

Step5: using While loop for identify the distance of mynewone is lower than the tolerance.

```
while self.euclidean_distance(goal_pose) >= distance_tolerance:
```

Step6: The end of the loop the turtle can stopped. After the movement is over we can stop the robot.

```
 # Stopping our robot after the movement is over.
vel_msg.linear.x = 0
  vel_msg.angular.z = 0
  self.velocity_publisher.publish(vel_msg)
```

Step7: Finally call the function and created the object x and the type of TurtleBot.

```
if __name__ == '__main__':
  try:
```

```
        x = TurtleBot()
      x.move2goal()
  except rospy.ROSInterruptException:
      pass
```

Step8: The code will be tested are following given command:

1. Open the terminal to run the below command.

$ roscore

$ rosrun turtlesim turtlesim_node

After execute the command and open the turtlesim windows open.

After finishing the process, next we open the new window. The below command are following to execute the code.

Open the new terminal to run the below code.

$ rosrun turtlesim_cleaner gotogoal.py

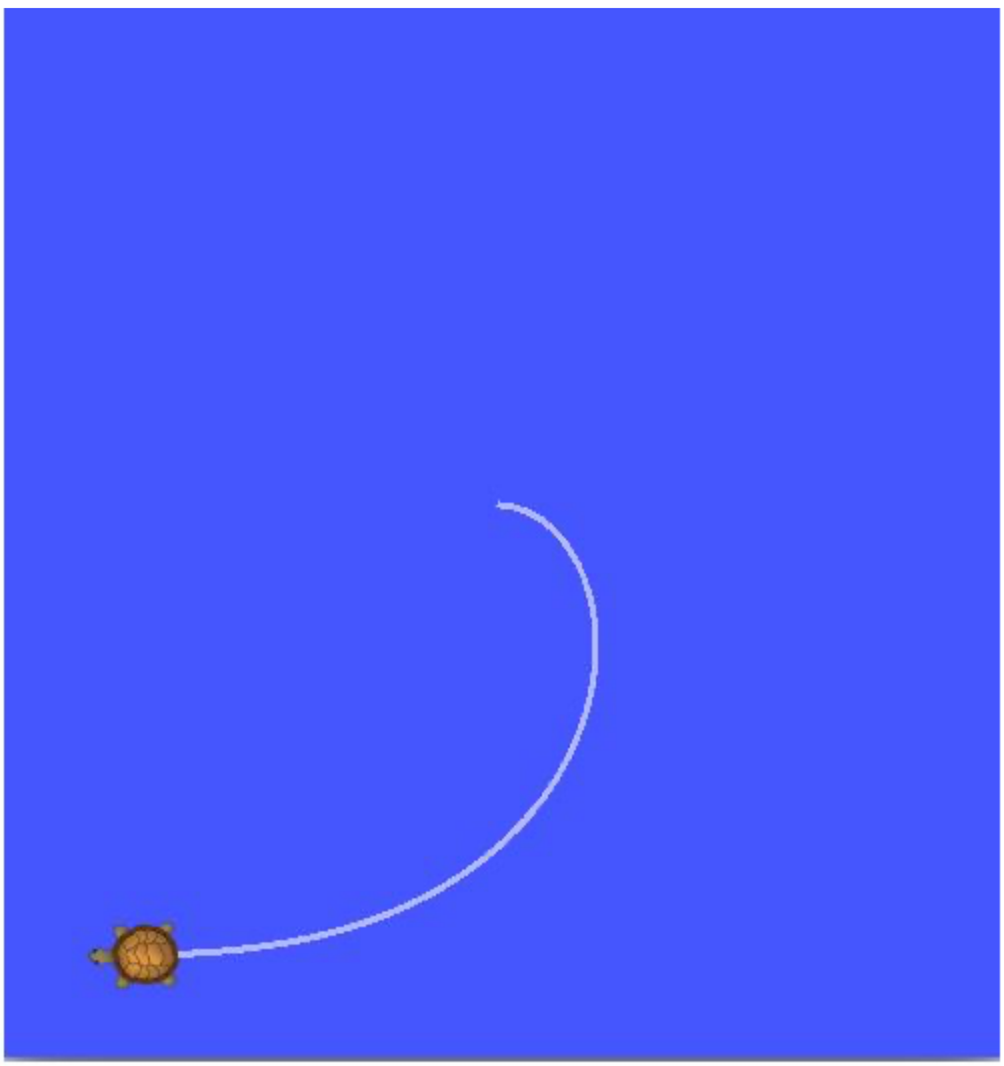
After the run the above code . Assign the input values to the other terminal. To movie the turtle. The turtle is named as the like the robots.

rosrun turtlesim_cleaner gotogoal.py

Set your x goal: 1

Set your y goal: 1

Set your tolerance: 0.5



Run the python code step by step Process:

Step1: create a project in your specific IDE

Step2: create a code and also create the exact package for this code.

Step3: Load the source file .The source file named as .py extension

Step4: After the load the file and compile the package.

Step5: Finally run the python code.

7. Conclusion

In this project the opto force tactile sensor as well as the kinova robot was integrated successfully. For that the detailed research on the robot system was conducted successfully. Based on the conducted study the required details about the hardware as well as software components are gathered. The detailed study about the kinova robot arm was successfully completed. The hardware components as well as the interfacing methods of the kinova arm are detail analyzed. Then the two PCB design was developed. Here the first PCB was responsible for establishing the connection between the sensors as well as the robot arm. The second PCB was responsible for establish the connection between the robot arm to the pc. After the successful completion of the designing part, all the components are assembled based on the proposed design.

References

Adamatzky, A. (2014). Tactile Bristle Sensors Made With Slime Mold. *IEEE Sensors Journal*, *14*(2), 324-332. doi: 10.1109/jsen.2013.2283192

Aggarwal, A., & Kirchner, F. (2014). Object Recognition and Localization: The Role of Tactile Sensors. *Sensors*, *14*(2), 3227-3266. doi: 10.3390/s140203227

Alfadhel, A., &Kosel, J. (2015). Tactile Sensors: Magnetic Nanocomposite Cilia Tactile Sensor (Adv. Mater. 47/2015). *Advanced Materials*, *27*(47), 7896-7896. doi: 10.1002/adma.201570326

Alsina, M., Escudero, F., Margalef, J., &Luengo, S. (2007). Tactile Fabric Panel in an Eight Zones Structure. *Sensors*, *7*(11), 2953-2969. doi: 10.3390/s7112953

Asaga, E., Takemura, K., Maeno, T., Ban, A., &Toriumi, M. (2013). Tactile evaluation based on human tactile perception mechanism. *Sensors And Actuators A: Physical*, *203*, 69-75. doi: 10.1016/j.sna.2013.08.013

Basics, C. (2018). How to Design a PCB Layout - Circuit Basics. Retrieved from http://www.circuitbasics.com/make-custom-pcb/

Breakthrough in programming with new ActiveRobot™. (2001). *Industrial Robot: An International Journal*, *28*(2). doi: 10.1108/ir.2001.04928bad.004

Calinon, S. (2009). *Robot programming by demonstration*. Lausanne: EPFL Press.

Campusano, M., &Fabry, J. (2017). Live Robot Programming: The language, its implementation, and robot API independence. *Science Of Computer Programming*, *133*, 1-19. doi: 10.1016/j.scico.2016.06.002

Clarkson, S. (2000). Design Tools—PCB Design & Make. *Electronics Education*, *2000*(2), 5-5. doi: 10.1049/ee.2000.0032

cplusplus.com - The C++ Resources Network. (2018). Retrieved from http://www.cplusplus.com/

DELMIA™ Products for Performance - DassaultSystèmes®. (2018). Retrieved from https://www.3ds.com/products-services/delmia/products/

Design Your Own PCB. (2018). Retrieved from https://www.jameco.com/Jameco/workshop/Video/design-your-own-printed-circuit-board.html

FANG, C., & DING, X. (2012). Anthropomorphic Arm Kinematics Oriented to Movement Primitive of Human Arm Triangle. *Robot*, *34*(3), 257. doi: 10.3724/sp.j.1218.2012.00257

FORTRAN | computer language. (2018). Retrieved from https://www.britannica.com/technology/FORTRAN

Fritzing. (2018). Retrieved from http://fritzing.org/learning/tutorials/designing-pcb/

Gong, D., He, R., Yu, J., &Zuo, G. (2017). A Pneumatic Tactile Sensor for Co-Operative Robots. *Sensors*, *17*(11), 2592. doi: 10.3390/s17112592

Hashim, I., Kumamoto, S., Takemura, K., Maeno, T., Okuda, S., & Mori, Y. (2017). Tactile Evaluation Feedback System for Multi-Layered Structure Inspired by Human Tactile Perception Mechanism. *Sensors*, *17*(11), 2601. doi: 10.3390/s17112601

Hughes, C., & Hughes, T. *Robot programming*.

Inc., Z. (2018). PCB Design Solutions | Zuken. Retrieved from https://www.zuken.com/en/products/pcb-design

Institute of Electrical and Electronics Engineers. (2001). *Colloquium on "Robot Sensors"*. Piscataway, NJ.

Jefferis, D. (2007). *Robot warriors*. New York: Crabtree Pub. Co.

Jung, Y., Lee, D., Park, J., Ko, H., & Lim, H. (2015). Piezoresistive Tactile Sensor Discriminating Multidirectional Forces. *Sensors*, *15*(10), 25463-25473. doi: 10.3390/s151025463

Kineo CAM: teachingless programming. (2012). *Industrial Robot: An International Journal*, *39*(6). doi: 10.1108/ir.2012.04939faa.007

Kinova Robotics - Robotics Business Review. (2018). Retrieved from https://www.roboticsbusinessreview.com/listing/kinova-robotics/

Martinez-Hernandez, U. (2015). Tactile Sensors. *Scholarpedia*, *10*(4), 32398. doi: 10.4249/scholarpedia.32398

Me, A., Vollom, D., Song, J., Burre, S., CM, S., &kumar, p. et al. (2018). PCB Design - How To Create Circuit Boards - Build Electronic Circuits. Retrieved from https://www.build-electronic-circuits.com/pcb-design/

Montaño, A., &Suárez, R. (2018). Manipulation of Unknown Objects to Improve the Grasp Quality Using Tactile Information. *Sensors*, *18*(5), 1412. doi: 10.3390/s18051412

Mytum-Smithson, J. (2007). Simulation and off-line programming. *Industrial Robot: An International Journal*, *34*(5). doi: 10.1108/ir.2007.04934eag.001

Newman, W. *A systematic approach to learning robot programming with ROS*.

OC Robotics provides snake-arm robot demonstration for Sellafield Ltd. (2012). *Industrial Robot: An International Journal*, *39*(1). doi: 10.1108/ir.2012.04939aaa.005

Off-line programming trims time and cost of routing application. (2004). *Industrial Robot: An International Journal*, *31*(1). doi: 10.1108/ir.2004.04931aaf.002

Owen-Hill, A. (2018). What is the Best Programming Language for Robotics?. Retrieved from https://blog.robotiq.com/what-is-the-best-programming-language-for-robotics

PCB Design, 7. (2018). 7 Fatal Mistakes to Avoid on Your PCB Design | Make:. Retrieved from https://makezine.com/2016/08/15/7-fatal-mistakes-avoid-printed-circuit-board-pcb-design/

Pressure profile systems enters OEM market with new line of digital tactile sensors: first capacitive tactile sensors with direct digital output. (2006). *Sensor Review*, *26*(3). doi: 10.1108/sr.2006.08726caf.004

Pugh, A. *Robot sensors*. Bedford: IFS (Publications).

RoboMind.net - Introduction. (2018). Retrieved from http://www.robomind.net/en/introduction.htm

Robot control and programming. (2006). *Industrial Robot: An International Journal*, *33*(6). doi: 10.1108/ir.2006.04933fag.001

Robot Library | RoboDK. (2018). Retrieved from https://robodk.com/library

Romeo, R., Oddo, C., Carrozza, M., Guglielmelli, E., &Zollo, L. (2017). Slippage Detection with Piezoresistive Tactile Sensors. *Sensors*, *17*(8), 1844. doi: 10.3390/s17081844

Roth, D. (2004). *Robot Programming*. [S.I.]: McGraw-Hill Education.

Trujillo-León, A., & Vidal-Verdú, F. (2014). Driving Interface Based on Tactile Sensors for Electric Wheelchairs or Trolleys. *Sensors*, *14*(2), 2644-2662. doi: 10.3390/s140202644

Ultra lightweight robotic arm. (2018). Retrieved from https://www.kinovarobotics.com/en/products/robotic-arm-series/ultra-lightweight-robotic-arm

Yuan, W., Dong, S., &Adelson, E. (2017). GelSight: High-Resolution Robot Tactile Sensors for Estimating Geometry and Force. *Sensors*, *17*(12), 2762. doi: 10.3390/s17122762

(2018). Retrieved from http://www.undergraduatelibrary.org/system/files/985.pdf

arm, I. (2018). Inter-processor communication for robotic arm. Retrieved from https://robotics.stackexchange.com/questions/736/inter-processor-communication-for-robotic-arm

Interfacing Serial Bluetooth Modem with Computer using ATmega16 | EngineersGarage. (2018). Retrieved from https://www.engineersgarage.com/contribution/expert/serial-bluetooth-modem-interfacing-using-atmega16

Kumar. (2018). RF based Wireless Robot using 8051 Microcontroller. Retrieved from https://www.slideshare.net/RahulKumar1128/rf-based-wireless-robot-using-8051-microcontroller

Sci-Hub: устраняя преграды на пути распространения знаний. (2018). Retrieved from https://sci-hub.tw/10.1109/ICECENG.2011.6057673