IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)

An International Open Access, Peer-reviewed, Refereed Journal

INTELLIGENT GOODS CARRIER ROBOT

[1] P. Kamal, [2] Upputuri Hari Babu

[1] B.Tech Student, QIS College of Engineering and Technology, Ongole, Andhra Pradesh, India, Pin : 523272

[2] Professor, QIS College of Engineering and Technology, Ongole, Andhra Pradesh, India, Pin : 523272

Abstract

Intelligent robots can make work safer and satisfying the customer needs as per the requirement. Robots are assuming an increasing range of jobs that are dangerous for humans, such as cleaning toxic or infected and high temperature applications like fire hazards etc. The Robots has the potential for share tasks or processes with workers, taking on those parts of the task or process that are unergonomic and repetitive, such as lifting, fetching and carrying. The main objective of this project is to minimize the total transportation time and effort to carry the loads very easily and conveniently to the required destination securely. Our approach has been implemented and evaluate in simulation. It can significantly reduce the human effort.

Keywords: Robot, Manipulator, Sensors, Actuators, End Effectors, Controllers and Software

1. INTRODUCTION

The Robot is a reprogrammable multifunctional manippulator designed to move material, parts, tools or specilized devices through various programmed motions for the performance of a variety of tasks.

Robotics is an interdisciplinary integrative field, at the joining together of several areas, ranging from mechanical and electrical engineering to control theory and computer science, with recent extensions toward material physics, bioengineering or cognitive sciences. The Robotics intersection is very rich. It covers issues such as:

- Deliberate action, planning, acting, monitoring and goal reasoning.
- Perceiving, modeling and understanding open environments.
- Interacting with human and other robots.
- Learning models required by the above functions.
- Integrating these functions in an adaptable and resilient architecture.

1.1 Robot components

1. Manipulator or Rover : Main body of robot (Links,joints, other structural elements)
2. End effectors : The part that is connected to the last joint (hand) of a manipulator.
3. Actuators : Muscles of the manipulators (servomotors, stepper motor, Pneumatic and Hydraulic cylinder)
4. Sensors: To collect information about the internal state of the robot or to communicate with the outside environment.
5. Controller : It controls and coordinates the motion of the actuators.
6. Processor: It is the brain of the robot. It calculates the motions and the velocity of the robots joints etc.
7. Software : Operating system, robotic software and the collection of routines.

1.2 The purpose of Robots

The robots are used for the following tasks

Dirty tasks

Repetitive tasks

Dangerous tasks

Impossible tasks

Robots assisting the handicapped

Can operate equipments at much higher precision than humans

Cheaper on a long term basis

2. LITERATURE REVIEW

The goods transportation system is an important application for autonomous mobile robot research, which can be used in various environments, such as warehouses, factories, container ports, or hospitals. Generally, how to complete the transportation mission with high efficiency and low cost is the first priority. On the one hand, in contrast to single robot, using a collaborative team of robots has the potential to accomplish the transportation mission more efficiently. **Tomio et al. [1]** designed a carrier mobile robot to work in official and domestic environments. The working path of the robot is defined by a black line in the environment and the robot is able to follow this pre-defined black line, like a line following robot, to move between source and destination. In this robot has been used for carrying small objects such as a cup and official letters between the rooms of manager and secretary. The different carrier robots were designed for agricultural applications, in which the robots are capable of carrying heavy agricultural tools that cannot be easily carried by farmers. In addition, depending on the sensors and processors embedded on the robot, the robot could collect some special information from the agricultural land **[2], [3]. Alami et al. [4]** reported a general concept for the control of a large fleet of autonomous mobile robots which has been developed, implemented and validated in the framework of MARTHA (Multiple Autonomous Robots for Transport and Handling Applications). They proposed an approach called plan-merging paradigm for multi-robot cooperation, which has been tested in both simulation and real world. **Vaughan et al. [5]** proposed a method (LOST) that enables a team of robots to navigate between places of interest in an initially unknown environment using a trail of landmarks. They applied this method as an example "resource transportation" task, in which multiple autonomous robots find and repeatedly traverse a path in an unknown environment between a known "home" and a supply of resource at an initially unknown position. **Tang and Parker [6]** described an approach for automatically synthesizing task solutions for heterogeneous multi-robot teams which is called Asymetrix. This approach is built upon representation and information invariants theories; it enables the robot team to dynamically connect plans within and across robots to accomplish a task. They also validated this approach in two different scenarios: multi-robot transportation and multi-robot box pushing. **Wawerla and Vaughan [7]** described two task allocation strategies for a multi-robot transportation system. One is based on a centralized planner that uses domain knowledge to solve the assignment problem in linear time. The other enables individual robots to make individual task allocation decisions using only locally obtainable information and single value communication. They used the energy expended by robots as performance evaluation standard. The computational complexity of these two strategies is small, but the performance is good. In contrast with previous research work, the basic object of our investigation is goods transportation in the real world with closer physical model, including grip action, drop action, and security of goods in transport.

3. WORKING PRINCIPLE OF LOAD CARRYING ROBOT

3.1 WORKING PRINCIPLE OF ROBOT

Our aim of designing the robot is to follow the person who is the owner of wearable for follower robot, following in straight path is simple by using IR beams as guide but to track right and left turn proper differential encoder has to be used which can be sensed by the robot and robot can take decision take right/left turn or go straight. IR proximity sensor is used detect obstacles and avoid collision. When the person required to stop robot at any situation it will stop by pressing stop button and all these decision are to be carried out by intelligent person following robot.

3.2 BLOCK DIAGRAM OF THE RECEIVER/ TRANSMITTER UNITS

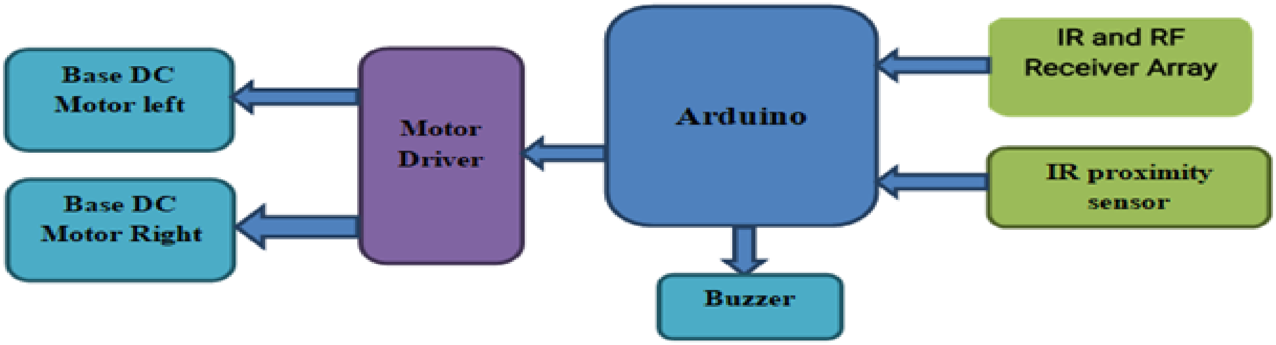


Figure:3.1 Block diagram of the Transmitter

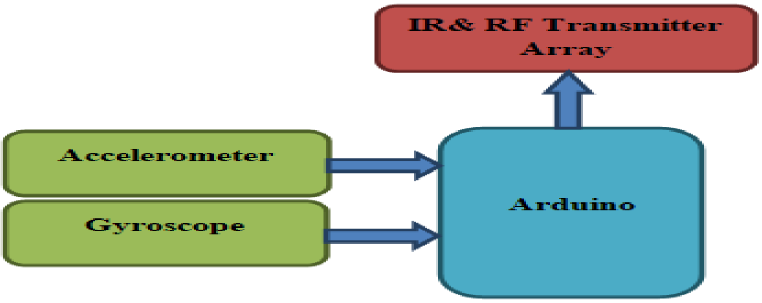


Figure:3.2 Block diagram of the Receiver

3.3 PROGRAMMING

The UNO can be programmed with the Arduino Software (IDE). Select Arduino/Genuino UNO from the Tools - Board menu (according to the microcontroller of the board).

The ATmega328 on the UNO comes preprogrammed with a boot loader that allows the user to upload new code to it without the use of an external hardware programmer. It communicates using the original STK500 protocol.

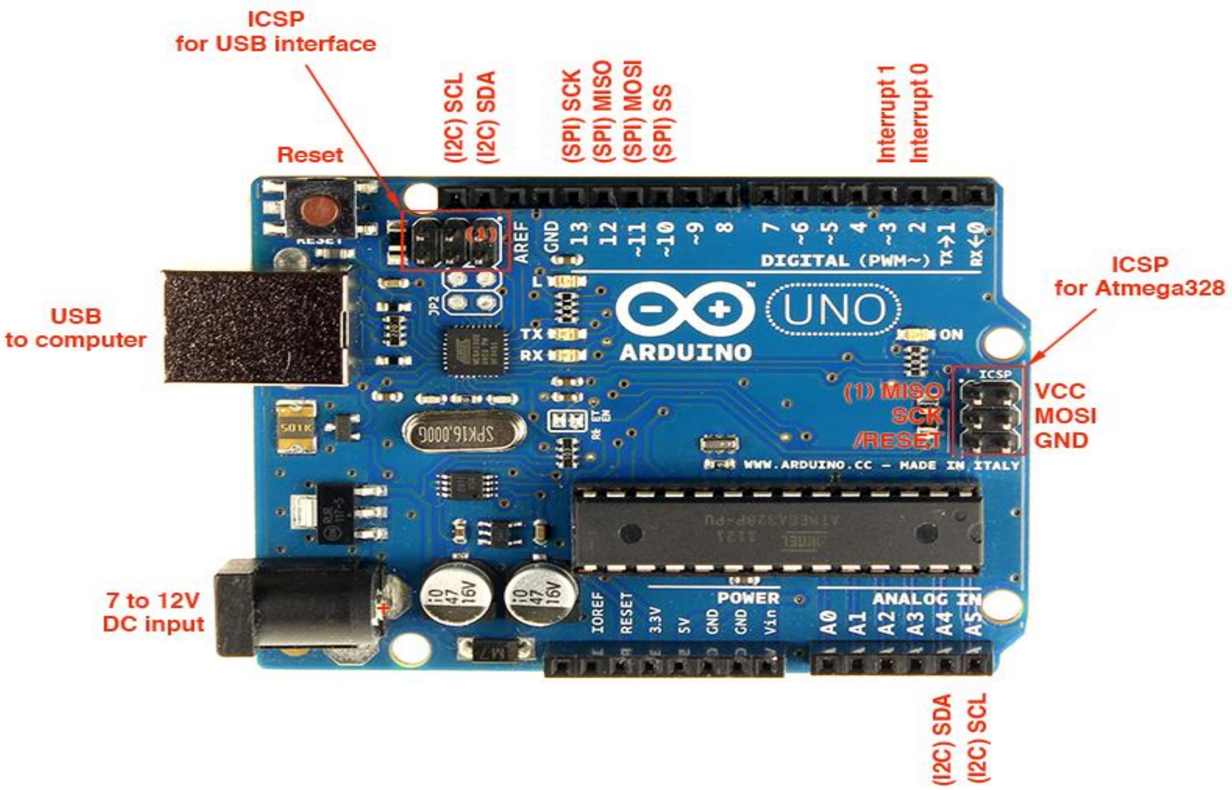


Figure: 3.3 Programming Arduino

3.3.1 Circuit Board of Arduino UNO

Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. The software, too, is open-source, and it is growing through the contributions of users worldwide.

UNO means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The UNO board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform; for an extensive list of current, past or outdated boards see the Arduino index of boards.

Figure: 3.4 Circuit Board of Arduino UNO

Arduino development board can sense the environment by receiving input from different sensors and affects its surroundings by controlling motors, lights and other actuators. The microcontroller on the board is programmed using the Arduino programming language.

3.4 CC2500 SIMPLIFIED BLOCK DIAGRAM

The RF transceiver is integrated with a highly configurable baseband modem. The modem supports various modulation formats and has a configurable data rate up to 500 kbps. The communication range can be increased by enabling a Forward Error Correction option, which is integrated in the modem.CC2500, provides extensive hardware support for packet handling, data buffering, burst transmissions, clear channel assessment, link quality indication and wake-on-radio.

The main operating parameters and the 64-byte transmit/receive FIFOs of CC2500 can be controlled via an SPI interface. In a typical system, the CC2500 will be used together with a microcontroller and a few additional passivecomponents.CC2500 is part of Chipcon’s 4th generation technology platform based on 0.18 μm CMOS technology.

With the typical current consumption of 13.3mA in receiving mode, 11.1mA in transmitting mode, and 400nA in sleep mode, the low-power feature of CC2500 is obviously shown. In addition, Wake-On-Radio (WOR) and fast start-up time which is generally 240s from sleep mode to receiving or transmitting mode contribute to the low-power feature too.

Besides the Wake-On-Radio mentioned above, CC2500 also supports packet handling, data buffering, a few modulation schemes, link quality indication, and some other functions. Two 64-byte FIFOs, for transmitting and receiving, respectively, are included in CC2500 and can be controlled via an SPI interface.

The first and foremost basic type of topology is the star topology, in which each node maintains a single, direct communication path with the gateway. This topology is simple but restricts the overall distance that a network can achieve. To increase the distance a network can cover, one can implement a cluster or tree topology. In cluster architecture, each node maintains a single communication path to the gateway but can use other nodes to route its data along that path. The drawback of this topology is that the nodes that depend on the router lose their communication paths to the gateway if the disconnection of router node exists.

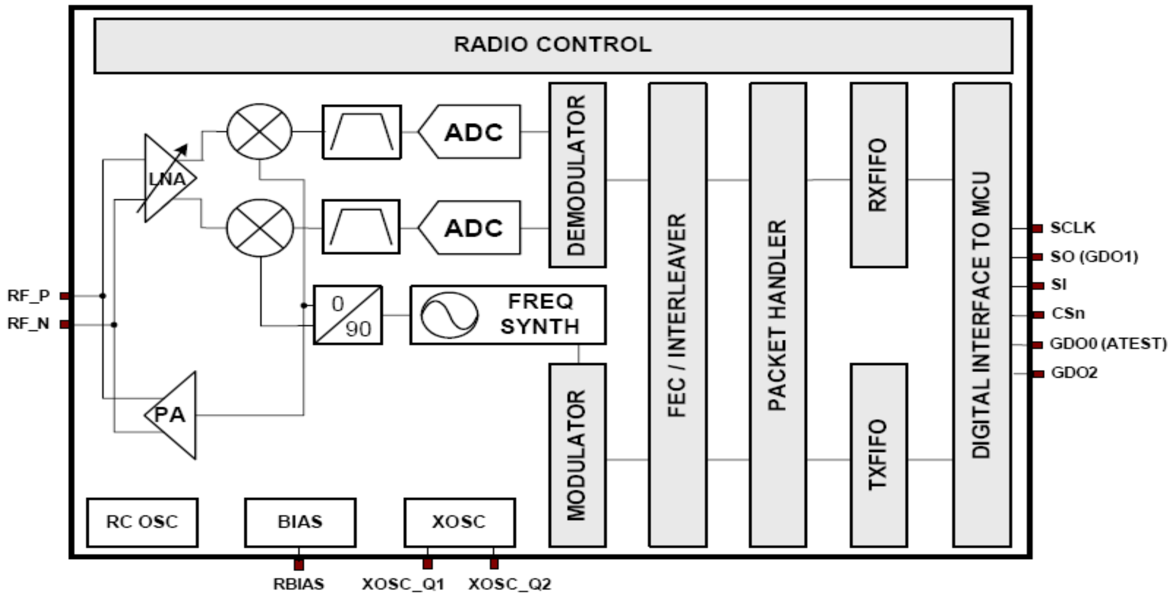


Figure: 3.5 CC2500 Simplified Block Diagram

Figure 3.5 is simplified block diagram of CC2500. The low-IF receiver part leads the received RF signal through a low-noise amplifier (LNA), the process of down-converting to the intermediate frequency (IF), channel filters, and ADCs to be digitized. The transmitter part relies on the frequency synthesizer with the frequency generated by the crystal oscillator. SO, SI, CSn, and SCLK compose the 4-wire SPI serial interface to configure the CC2500.

4. PRO/E SOFTWARE

Pro/ENGINEER is a computer graphics system for modeling various mechanical designs and for performing related design and manufacturing operations. The system uses a 3D solid modeling system as the as the core, and applies the feature-based, parametric modeling method. In short, Pro/ENGINEER is a feature -based, parametric solid modeling system with many extended design and manufacturing applications manufacturing applications.

The basic functionality of Pro/ENGINEER is broken into four major areas:

- Part Modeling and Design
- Assembly Modeling and Design
- Design Documentation (Drawing Generation)
- General Functionality

5. DESIGN OF ROBOT STRUCTURE USING PRO/E

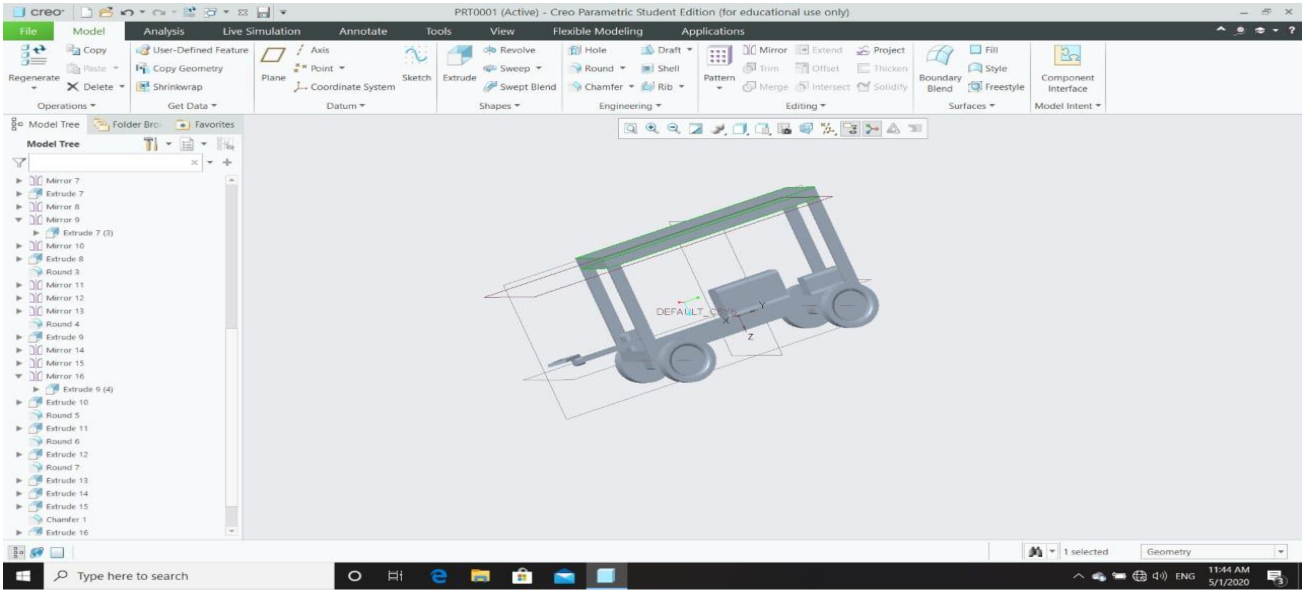


Figure: 5.1 Basic structure of Robot

5.2 Specification of Robot

Distance between two wheel centers: 30 cms

Height of the structure: 10 cms

Width of the base: 29 cms

Size of Square pipe: 1½ inch

Wheel Diameter: 7*4 cms

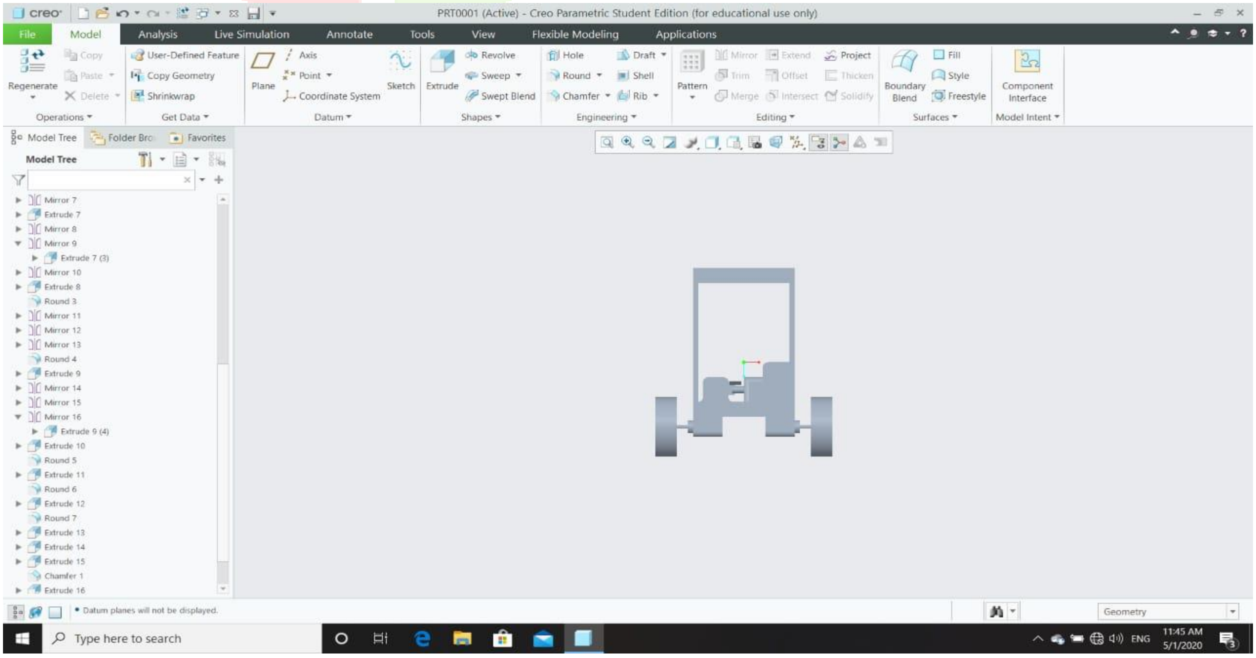
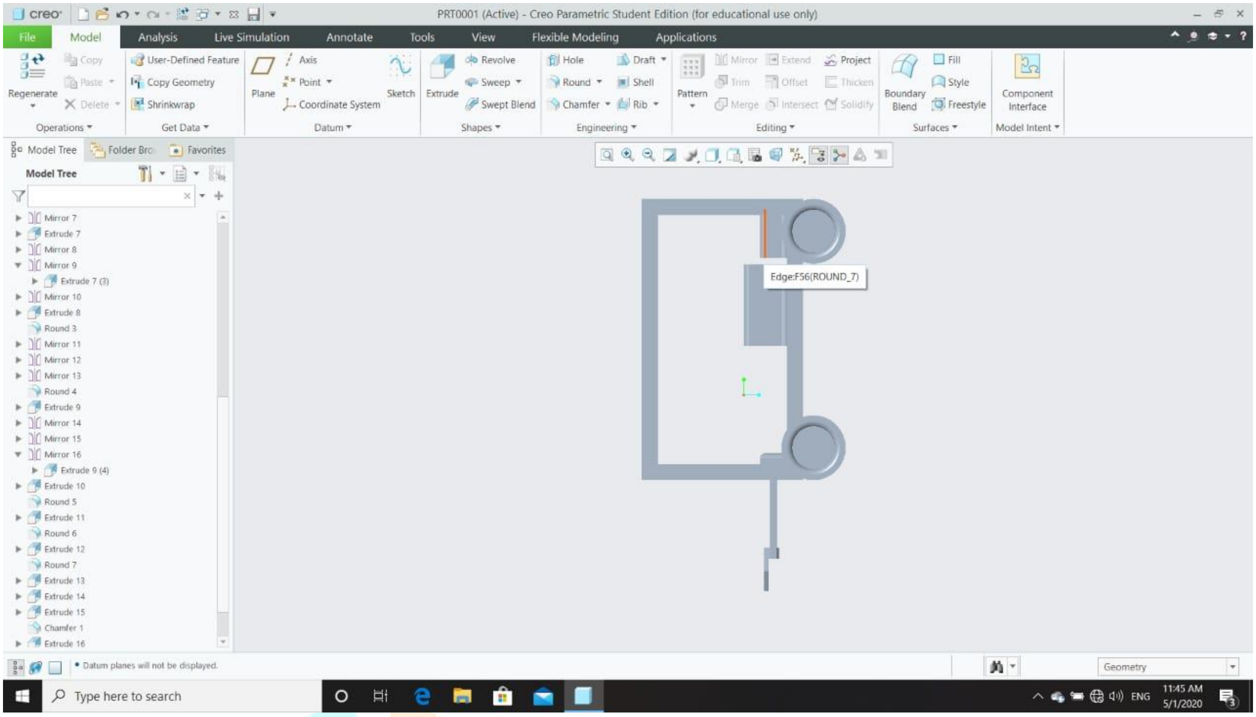
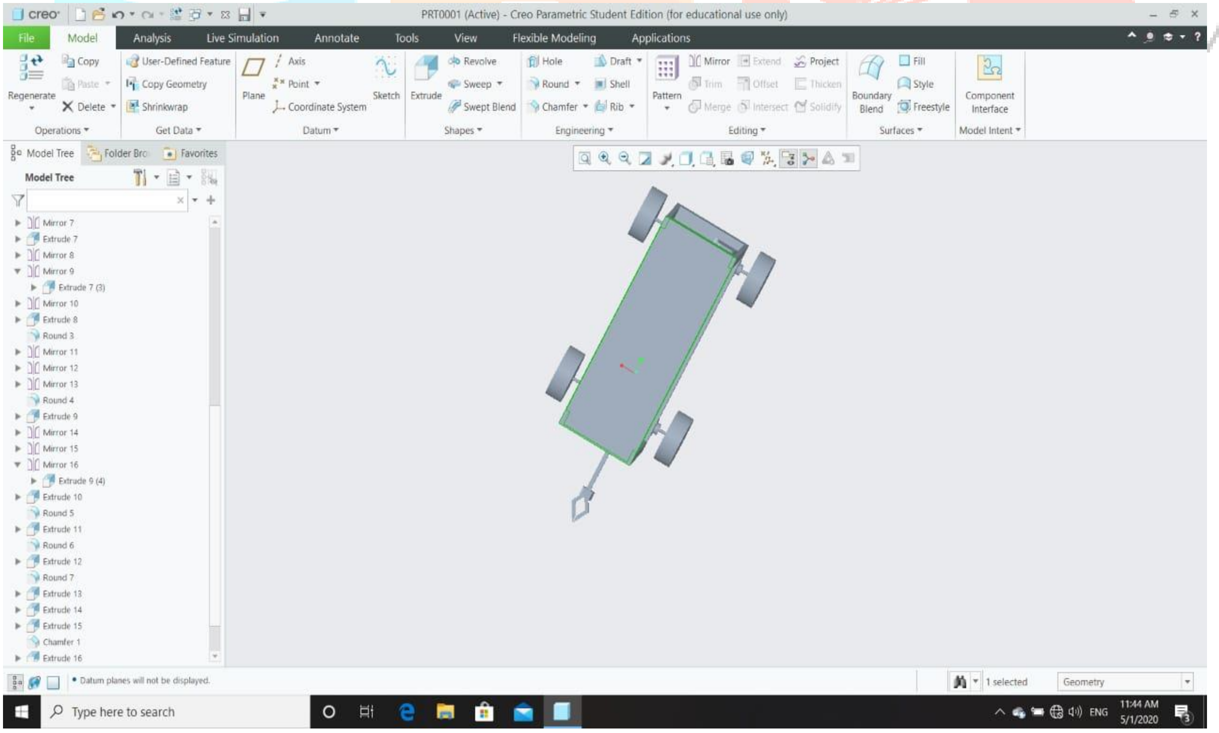


Figure: 5.2 Back View of Robot


PRT0001 (Active) - Creo Parametric Student Edition (for educational use only)

Datum planes will not be displayed.

Figure: 5.3 Side View of Robot


PRT0001 (Active) - Creo Parametric Student Edition (for educational use only)

Datum planes will not be displayed.

Figure: 5.4 Top View of Robot

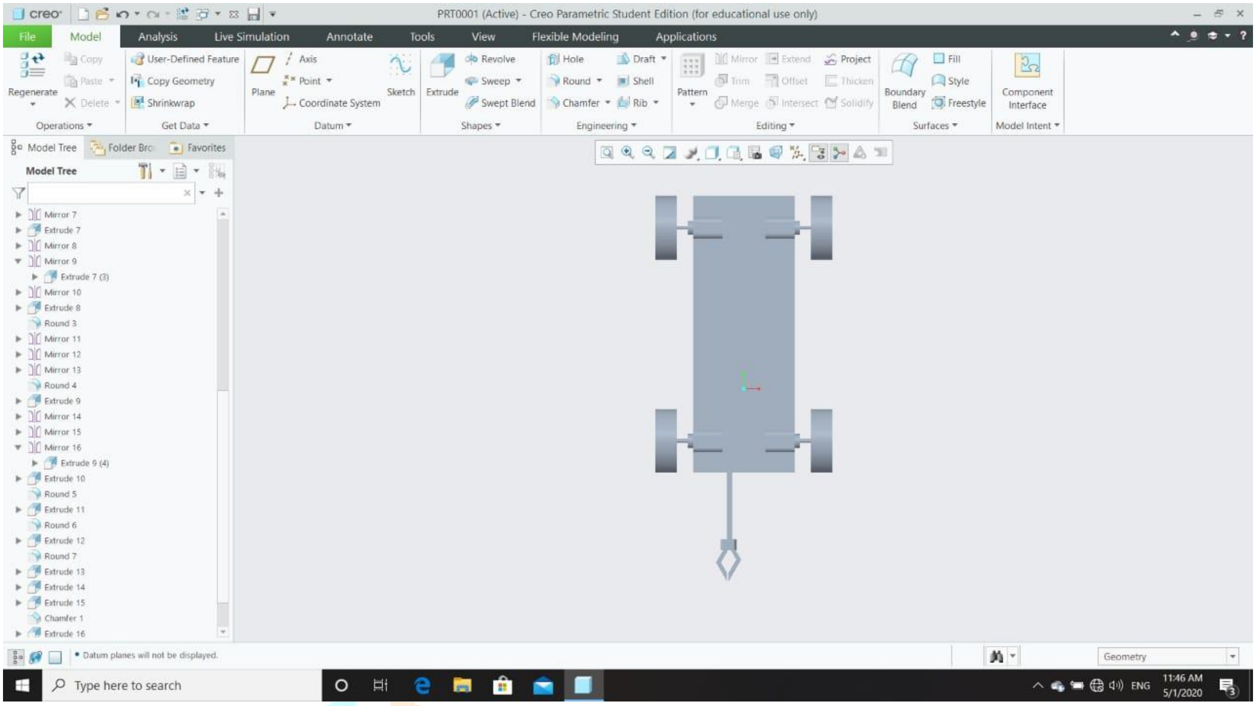


Figure: 5.5 Ground View of Robot

6. EMBEDDED C PROGRAMMING

Embedded C is perhaps the most popular languages among Embedded Programmers for programming Embedded Systems. There are many popular programming languages like Assembly, BASIC, C++ etc. that are often used for developing Embedded Systems but Embedded C remains popular due to its efficiency, less development time and portability. Before digging in to the basics of Embedded C Program, we will first take a look at what an Embedded System is and the importance of Programming Language in Embedded Systems.

Before going in to the details of Embedded C Programming language first know about the C Programming language. The C programming language, developed by Dennis Ritchie in the late 60’s and early 70’s, is the most popular and widely used programming language. The C programming language provided low level memory access using an uncomplicated compiler software that converts programs to machine code) and achieved efficient mapping to machine instructions. The C Programming Language became so popular that it is used in a wide range of applications ranging from Embedded Systems to super computers.

6.1 Programming Embedded Systems

As mentioned earlier, embedded systems consists of both hardware and software. If we consider a simple embedded system, the main hardware module is the processor. The Processor is the heart of the embedded system and it can be anything like a Microprocessor, Microcontroller, DSP, CPLD (Complex Programmable Logic Device) and FPGA (Field Programmable Gated Array).

All these devices have one thing in common: they are programmable i.e. we can write a program (which is the software part of the embedded system) to define how the device actually works.

Embedded software or Program allows hardware to monitor external events (Inputs) and control external devices (outputs) accordingly. During this process, the program for an embedded system may have to directly manipulate the internal architecture of the embedded hardware (usually the processor) such as timers, serial communications interface, interrupt handling, and I/O Ports etc. From the above statement, it is clear that the software part of an embedded system is equally important to the hardware part. There is no point in having advanced hardware components with poorly written programs (Software).

6.2 Factors for Selecting the Programming Language

The following are few factors that are to be considered while selecting the programming language for the development of embedded systems.

- **Size:** The memory that the program occupies is very important as Embedded Processors like Microcontrollers have a very limited amount of ROM.
- **Speed:** The programs must be very fast i.e. they must run as fast as possible. The hardware should not be slowed down due to a slow running software.
- **Portability:** The same program can be compiled for different processors.
- **Ease of Implementation**
- **Ease of Maintenance**
- **Readability**

7. ADVANTAGES & DISADVANTAGES OF LOAD CARRIER ROBOT

Advantages

- Speed.
- Consistency.
- Perfection.
- Productivity.
- Timely.

Disadvantages

- Need a supply of power.
- Cost is more for repair & maintenance.
- Can store large amounts of data but the storage, access, retrieval is not as effective as the human brain.

7. CONCLUSIONS

In this project design and fabrication of an intelligent load carrier robot has been carried out, for carrying loads in real-world environments such as industrial, domestic, hospital and official applications. The environment is allowed to have under certain static and dynamic obstacles. The navigation system of BarBot has two modes including learning and optimal path planning modes. In the learning mode, user can navigate the robot in the environment arbitrarily according to the desired task, and the robot can learn the user-defined paths. In optimal path planning mode, user can define the map and static obstacles of the environment in the HMI software of BarBot, and then the robot intelligently finds the shortest paths using an optimal path planning algorithm.

REFERENCES

[1] K. Tomio et al., "Designing of a Carrier Robot “CARREY” at Office," presented at the 6th Asian Design Conference, Japan, 2003.

[2] S. H. Nielsen, K. Jensen, A. Bogild, O. J. Jorgensen, N. Jacobsen, C. L. Jaeger-Hansen, "A low cost, modular robotics tool carrier for precision agriculture research," in 11th International Conference on Precision Agriculture, 2012.

[3] R. N Jorgensen, C. G.Sorensen, J. Maagaard, I. Havn, K. Jensen, H. Sogaard, "HortiBot: A system design of a robotic tool carrier for high-tech plant nursing," Agricultural Engineering International: the CIGR Ejournal. Manuscript ATOE 07 006. Vol. IX, July, 2007.

[4] R. Alami, S. Fleury, M. Herrb, F. Ingrand, and F. Robert, “Multi-robot cooperation in the MARTHA project,” IEEE Robotics and Automation Magazine, vol. 5, no. 1, pp. 36–47, March 1998.

[5] R. T. Vaughan, K. Stoy, G. S. Sukhatme, and M. J. Matari´c, “LOST: Localization-space trails for robot teams,” IEEE Transactions on Robotics and Automation, vol. 18, no. 5, PP. 796–812, October 2002.

[6] F. Tang and L. E. Parker, “ASyMTRe: Automated synthesis of multi-robot task solutions through software reconfiguration,” in Proceedings of the 2005 IEEE International Conference on Robotics and Automation (ICRA’05), Barcelona, Spain, April 2005, PP. 1501–1508.

[7] J. Wawerla and R. T. Vaughan, “A fast and frugal method for team task allocation in a multi-robot transportation system,” in Proceedings of the 2010 IEEE International Conference on Robotics and Automation (ICRA’10), Anchorage, AK, USA, May 2010, PP. 1432–1437.

[8] Robin R. Murphy, Introductions to AI Robotics, MIT Press.mit.edu, 2020, ISBN: 9780262133838.

[9] Nikolaus Correll, Introduction to Autonomous Robots, Create Space Independent Publishing Platform; 1st edition, 2014 ISBN-13: 978-1493773077.

[10] Ivan Bratko, Prolog : programming for Artificial Intelligence, 3rd edition, Pearson, 2002.