TRIBHUWAN UNIVERSITY
INSTITUTE OF ENGINEERING
PULCHOWK CAMPUS

A Project Report
On

AUTO OBJECT TRACKER

Submitted To:

Department of Electronics and Computer Engineering

Submitted By:

Milan Maharjan (072bex446)
Srijana Bhusal (072BEX440)
Sunidhi Amatya (072bex443)
Madhusudan Mainali (072BEX449)

ACKNOWLEDGEMENT

We would like to extend our sincere gratitude to lecturer Mr. **Dinesh Baniya Kshetri** (Electronics and Computer Engineering Department, Central Campus Pulchowk)for providing us an opportunity to do a Project and providing us with continuous support and suggestions during the course of searching the idea for our project.

We are deeply indebted to our seniors who encouraged us to study and work on this topic. We are highly obliged to them for their valuable support, insight and time without which completion of this proposal would have been impossible.

Finally, we must acknowledge our deep sense of gratitude to our friends, well-wishers and every individual for encouraging and supporting us in this initiative. And our words would be really less if we forget to mention our dearest mentors Mr. Sabin Acharya and Mr. Dil Raaj Gurung for their invaluable guidelines and mentorship. Finally all the suggestions that will be helpful for the further development of the project are highly welcomed.

ABSTRACT

Our approach here is to design something meaningful and useful with the knowledge we have about interfacing various electronics devices and explore their uses and working mechanism. This project is basically concerned with the development of a wirelessly controlled car with a feature to provide real time surveillance from the camera that is mounted on top of it.

The main objective of this project report is to provide a basic understanding of all the things that we came across during the course of our project development in a nutshell. This project of ours mainly focuses on the practical implementation of knowledge that we, the students of bachelor in electronics engineering, possess with a mix of some innovative ideas. In this project, we have successfully incorporated the use of AVR (Automatic Voltage Regulator) for vehicle control and designed a PCB(Printed Circuit Board). In addition to that, we have also used and interfaced Raspberry pi with its camera module for the live video streaming and object following tasks. Main goal behind our project was to create something that can be remotely accessed and controlled from a fair distance and make it perform something that is quite difficult for a human under certain circumstances.

TABLE OF CONTENTS

TOPICS **page no.**

1) Introduction ………………………………………………………………………………

2) Objectives………………………………………………………………………………………

3) Theory / Background ……………………………………………………………………….

4) Methodology……………………………………………………………………………………….

4.1) Design

4.2) Implementation

4.3) Block Diagram

4.4) Circuit Diagram

4.5) Testing Mechanism

4.6) Evaluation Mechanism

4.7) Required Hardware and Software Tools

5) Results………………………………………………………………………………………………

6) Discussion and Analysis ……………………………………………………………………..

7) Project Planning (Gantt Chart) ……………………………………………………………….

8) Budget Analysis …………………………………………………………………………………….

9) Conclusion ……………………………………………………………………………………………..

10) Appendices…………………………………………………………………………………………..

11) References…………………………………………………………………………………………..

1. INTRODUCTION

Being the students of electronics, we were ought to do a minor project and the subject we chose whose main criteria were to build something by making interface of any electronic device or component or more with the use of an AVR in collaboration with the receiver's circuits and do some image processing tasks and for that we used Raspberry PI. Fulfilling all the above mentioned criteria, we successfully built the AUTO OBJECT TRACKER. As the name suggests, it basically is a wirelessly controlled robotic car which is driven or controlled by the signals coming from the atmeag32 to the motor driving circuit after processing the image. For the live video streaming and image processing tasks, we use raspberry pi 3 and router forming a closed network.

2. OBJECTIVES

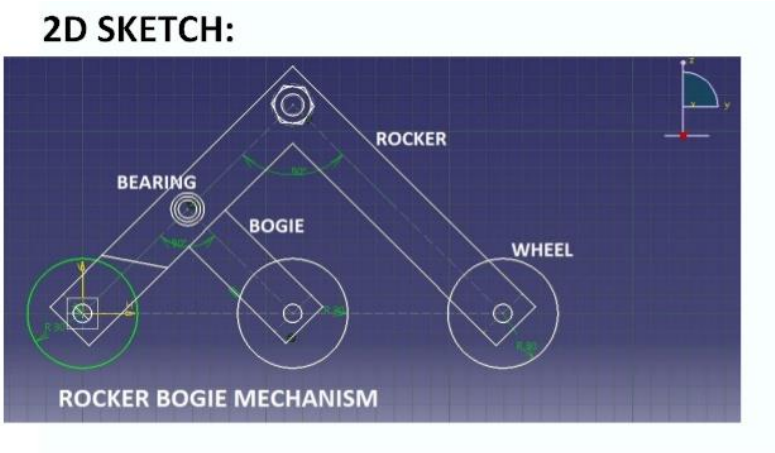
- to learn about various interfacing techniques for the interface of an AVR with other electronic components
- to use Raspberry pi for image processing(i.e to follow an object)
- to design a PCB

3. Theory/Background

We proposed a wireless tracker to take live video which could be accessed wirelessly. The details have been described below:

1. PHYSICAL DESIGN

   For suitable design, we have used rocker bogie mechanism for our car. It is a suspension arrangement used in Mars Rover for Mars Pathfinders and also in Mars Exploration Rover(MER).The main purpose of this system is to develop a highly stable suspension system capable of operating in multi terrain surfaces while keeping all the wheels in contact with the ground. Right and left rockers can independently climb different obstacles. Theoretically this mechanism can sustain tilt of about 50° inclination without tipping.



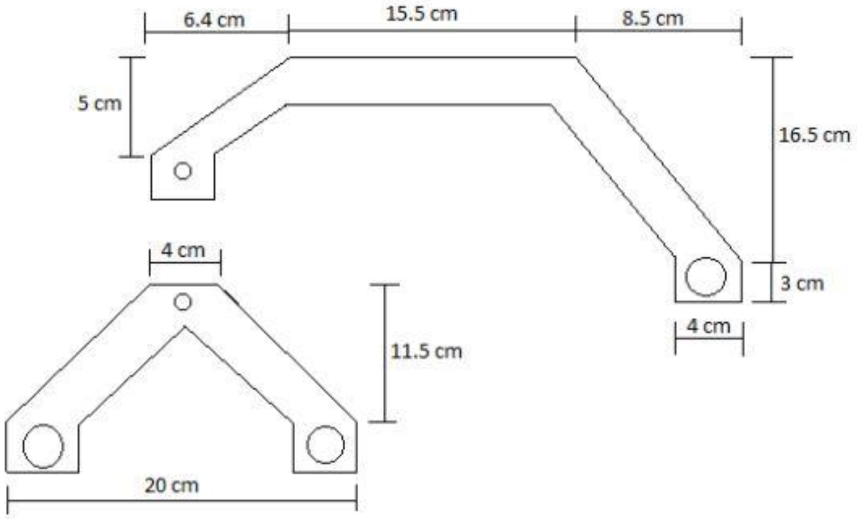


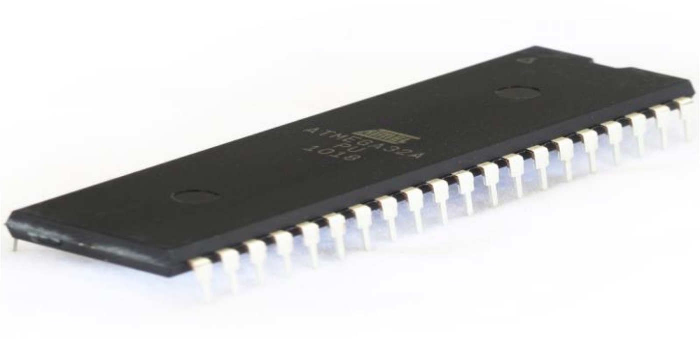
Fig: sketch 2D design for our car.

2. Manipulation of received data

The data received from raspberry pi now needs to be analyzed to take appropriate actions, i.e to run motors in desired directions. The data is analyzed and processed using Atmega32A which drives the motor driver L293D which in return drives the motors.

Components:

- **ATmega32A:**



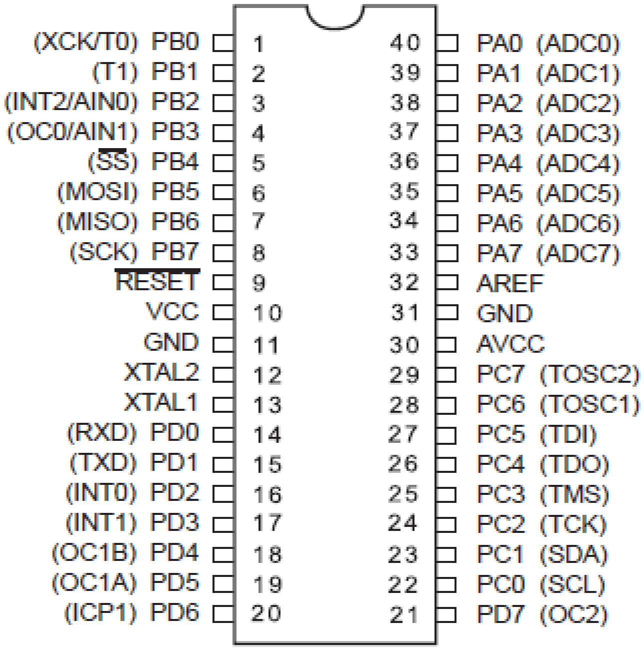


Fig: pin configuration

The Atmel® AVR® ATmega32A is a low-power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega32A achieves throughputs approaching 1 MIPS per MHz allowing the system designed to optimize power consumption versus processing speed.

Features

- High-performance, Low-power Atmel® AVR® 8-bit Microcontroller

- Advanced RISC Architecture
  - 131 Powerful Instructions – Most Single-clock Cycle Execution
  - 32 × 8 General Purpose Working Registers
  - Fully Static Operation
  - Up to 16MIPS Throughput at 16MHz
  - On-chip 2-cycle Multiplier

- High Endurance Non-volatile Memory segments
  - 32Kbytes of In-System Self-programmable Flash program memory
  - 1024Bytes EEPROM
  - 2Kbytes Internal SRAM
  - Write/Erase Cycles: 10,000 Flash/100,000 EEPROM
  - Data retention: 20 years at 85□C/100 years at 25□C(1)
  - Optional Boot Code Section with Independent Lock Bits

- In-System Programming by On-chip Boot Program

- True Read-While-Write Operation
  - Programming Lock for Software Security

- JTAG (IEEE std. 1149.1 Compliant) Interface
  - Boundary-scan Capabilities According to the JTAG Standard
  - Extensive On-chip Debug Support
  - Programming of Flash, EEPROM, Fuses, and Lock Bits through the JTAG Interface

- Atmel QTouch® library support
  - Capacitive touch buttons, sliders and wheels
  - Atmel QTouch and QMatrix acquisition
  - Up to 64 sense channels

- Peripheral Features
  - Two 8-bit Timer/Counters with Separate Prescalers and Compare Modes
  - One 16-bit Timer/Counter with Separate Prescaler, Compare Mode, and Capture Mode
  - Real Time Counter with Separate Oscillator
  - Four PWM Channels
  - 8-channel, 10-bit ADC

- 8 Single-ended Channels

- 7 Differential Channels in TQFP Package Only

- 2 Differential Channels with Programmable Gain at 1x, 10x, or 200x

– Byte-oriented Two-wire Serial Interface
– Programmable Serial USART
– Master/Slave SPI Serial Interface
– Programmable Watchdog Timer with Separate On-chip Oscillator
– On-chip Analog Comparator

- Special Microcontroller Features

– Power-on Reset and Programmable Brown-out Detection
– Internal Calibrated RC Oscillator
– External and Internal Interrupt Sources
– Six Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby

- I/O and Packages

– 32 Programmable I/O Lines
– 40-pin PDIP, 44-lead TQFP, and 44-pad QFN/MLF

- Operating Voltages

– 2.7V - 5.5V

- Speed Grades

– 0 - 16MHz

- Power Consumption at 1MHz, 3V, 25□C

– Active: 0.6mA
– Idle Mode: 0.2mA
– Power-down Mode: < 1µA

We have used the lower four pins of Port A (PA0-PA3) to input 4-bit data from HT12D into the microcontroller and the higher four pins are pulled up. The eight bit data at Port A is now compared with pre calculated values to determine a unique output to other ports. The output from Atmega is fed to motor driver L293D.

- **L293D:**

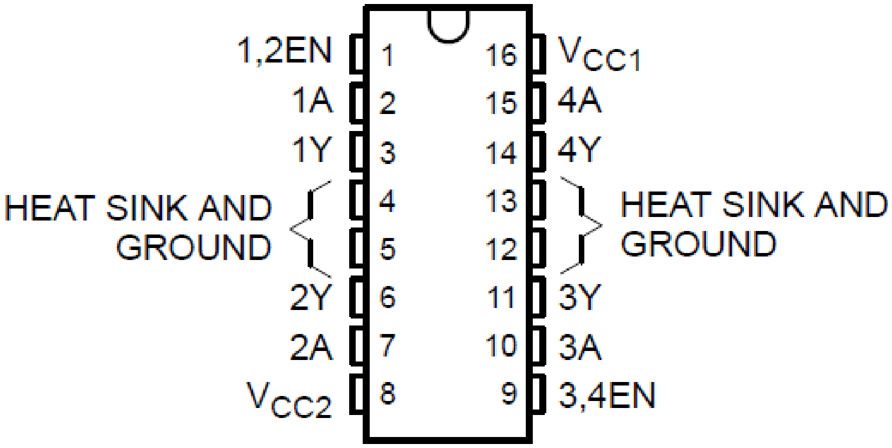


fig: pin configuration

The L293 and L293D devices are quadruple high-current half-H drivers. The L293 is designed to provide bidirectional drive currents of up to 1 A at voltages from 4.5 V to 36 V. The L293D is designed to provide bidirectional drive currents of up to 600-mA at voltages from 4.5 V to 36 V. Both devices are designed to drive inductive loads such as relays, solenoids, DC and bipolar stepping motors, as well as other high-current/high-voltage loads in positive-supply applications.

Each output is a complete totem-pole drive circuit, with a Darlington transistor sink and a pseudo- Darlington source. Drivers are enabled in pairs, with drivers 1 and 2 enabled by 1,2EN and drivers 3 and 4 enabled by 3,4EN.
The L293 and L293D are characterized for operation from 0°C to 70°C.

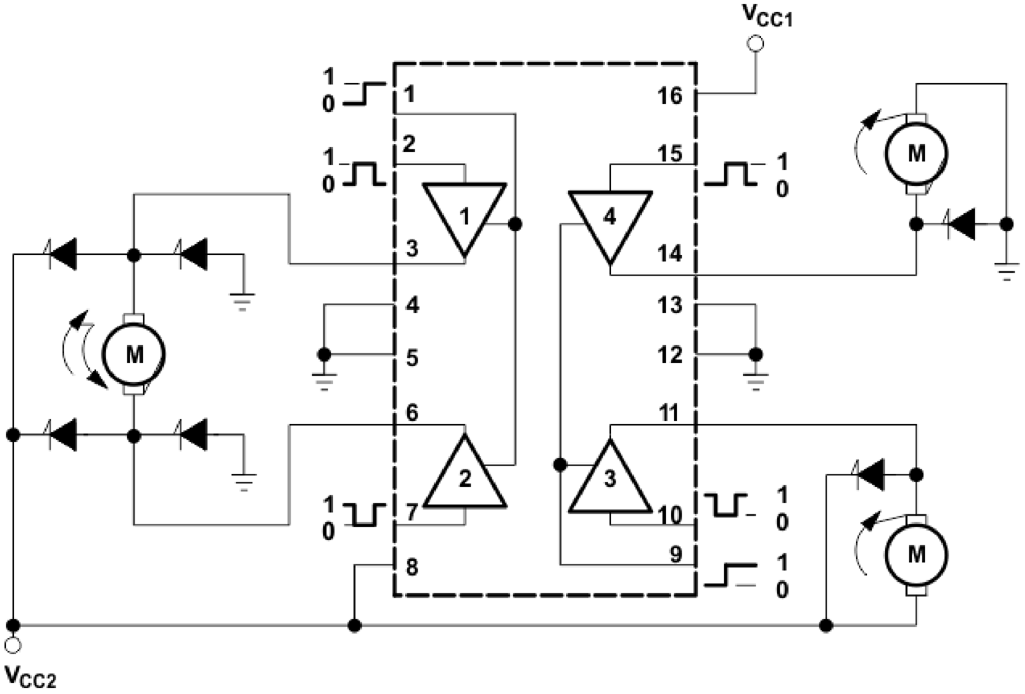
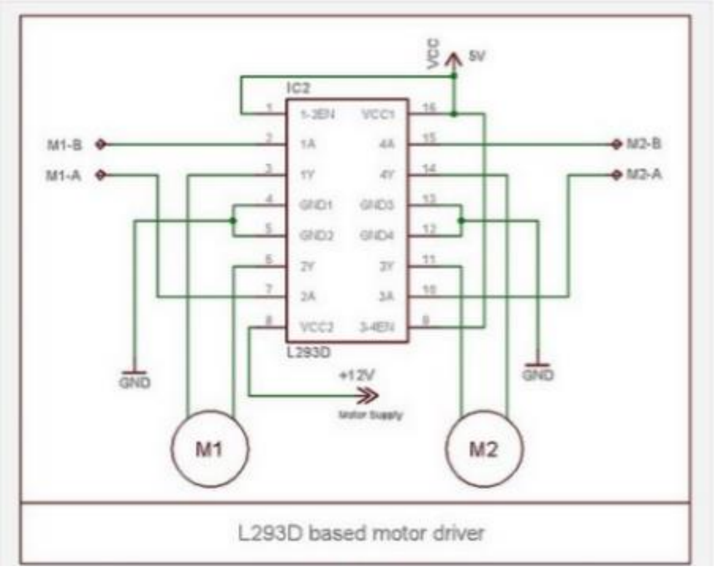


Fig: functional diagram

Features

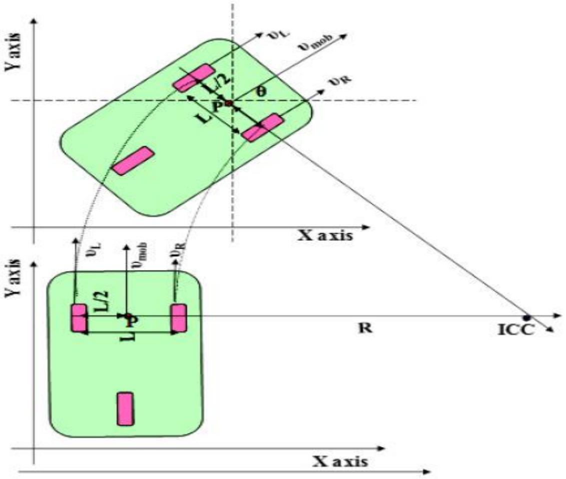
- Wide Supply-Voltage Range: 4.5 V to 36 V
- Separate Input-Logic Supply
- Internal ESD Protection
- High-Noise-Immunity Inputs

Three L293D motor drivers are used to drive the six dc motors placed to move six wheels of our car. The input signal for the input pins of L293D and for the enable pins are taken from the output ports of the microcontroller. Each motor driver having four output pins can drive two motors in both clockwise and counter clockwise direction. The pin diagram of L293D is given below:



PWM is used in Atmega and sent to enable pins of l293D to control speed of the car and differential drive motion used for changing directions.

DIFFERENTIAL DRIVE MOTION:



By changing the velocities of left and right wheel, ICC of rotation is changed and the wheels will follow different trajectories. This is called as differential drive motion. Angular velocity is given as

$$\omega = \frac{V_R - V_L}{L}$$

The curvature radius of trajectory is given as

$$R = \frac{V_R + V_L}{V_R - V_L} * \frac{L}{2}$$

Pulse width modulation (PWM) is used to vary velocities. PWM varies the duty cycle of the input pulse reaching the motors of the wheels.

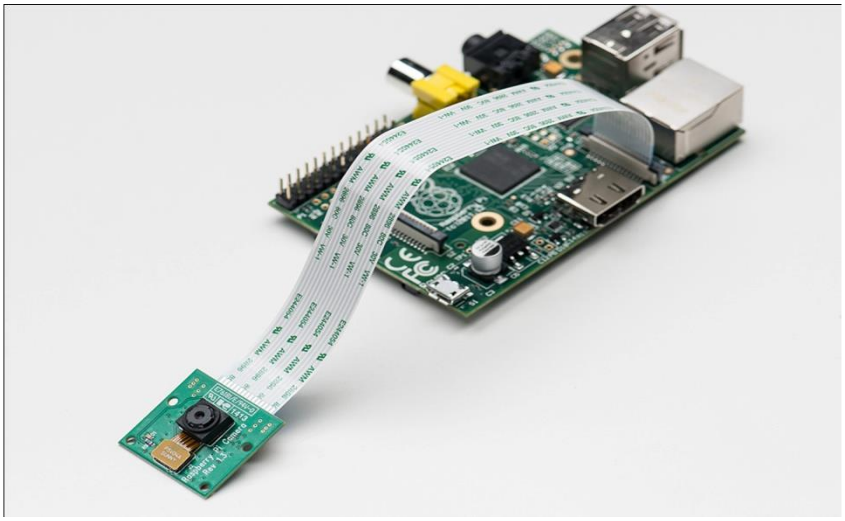
- **DC Motor:** A DC motor is any class of rotary electrical machines that converts direct current electrical energy into mechanical energy. The most common types rely on the forces produced by magnetic fields. Nearly all types of DC motors have some internal mechanism, either electromechanical or electronic to change the direction of current flow in part of motor.

Here, we have made use of 6 DC motors. One for each tyres which are 12 V motors of 350 Rpm. Tyre motors are driven and controlled by a L293D chip(described later).

- **Live video streaming:**

Fig: raspberry Pi 3

Raspberry Pi is a credit card sized computer that plugs into our computer and keyboard. It is a capable little computer which can be used in electronics projects, and for many of the things that our computer does, like spreadsheet, word processing, browsing the internet etc. It also plays high definition videos.



So the main information which is a video, can be remotely streamed is different ways. One of the best way is using Wi-Fi technology which is wireless. As video is huge information we need a hardware that is fast enough to capture the frames and send it wirelessly to the local machine. Our local machine (where we view the video) and the device should be connected wirelessly. Our best hardware platform is using no other Raspberry PI which is robust reliable, fast, cheap, and small in size. Raspberry Pi can handle huge amount of data and can fit our requirements. A newer version of Raspberry Pi has built in WIFI Technology which we can use to connect to our local router. So our hardware problem is tackled. Now what about controlling the raspberry pi remotely for video feedback? It can be using MATLAB, great software for program prototyping, robust and fast capable of performing complex processing like Image Processing, computer vision, etc. MATLAB can easily establish a connection to Raspberry Pi and access its hardware components. We will write programs in MATLAB that can remotely access the camera board of Raspberry Pi and stream it in local machine. Not only that we use image processing to extract information from the video and apply computer vision to the robot.

4. METHODOLOGY

4.1 DESIGN

Our system consists of a self-made printed circuit board, raspberry pi, and l293d as major components. We designed a PCB so that all of the components can be placed on a single board where power supply was also converted accordingly as per the requirement. In this way we were able to confine the arrangement into a compact box exposing almost nothing except for very few wires.

We used Atmega 32A in our project because we found it easy to prototype, work with, cheap and had plenty of resources to get started with it. Moreover, Atmega 32A was the most suitable choice for us in terms of performance and usability. Atmega 32A acted as interfacing device between the raspberry pi and l293d. However, we have also used router so that we can increase our range of working. The pi camera was used to capture current location of the device to be detected which sent the data to the Matlab from where the main algorithms were processed and compiled. L293d was used to drive motor according to the signals received from atmega32a which in turn were sent from raspberry pi after processing the current location of the object.

4.2 IMPLEMENTATION

During the implementation of the system, following steps were taken: Planning and Selection:

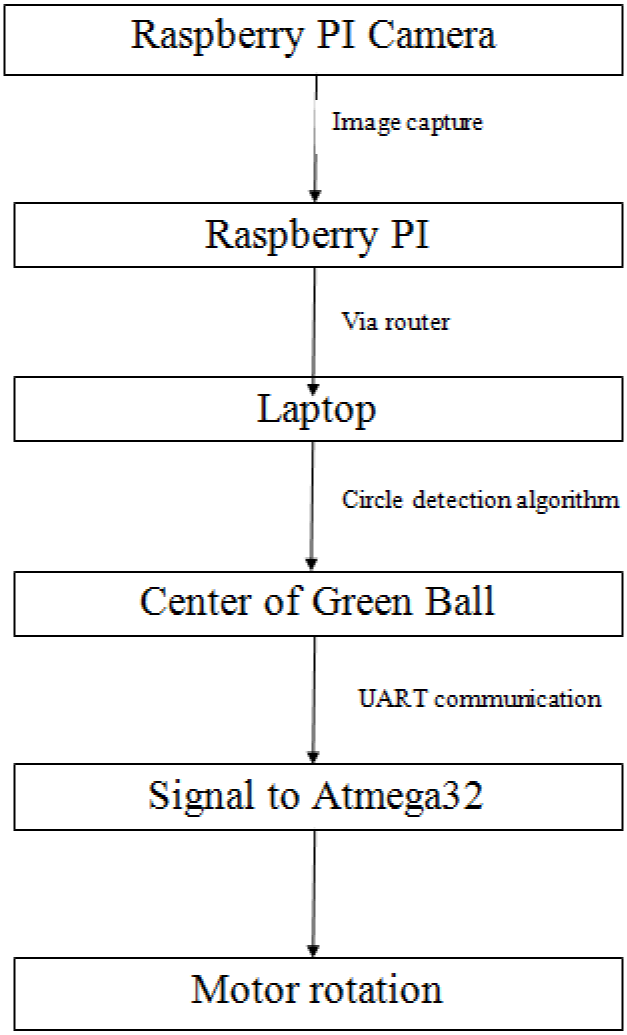
1. At first different ways of measuring the current location of the object was researched and we set the position of camera.
2. Further research was done in choosing device which could process the image of the object. We had options of both Matlab and opencv but we found Matlab to be easier for quick learning.
3. Next, we decided to use At mega 32A on a new pcb instead of using whole of the AVR board..
4. Since the control was needed within a larger area we settled on using the router too.

Execution:

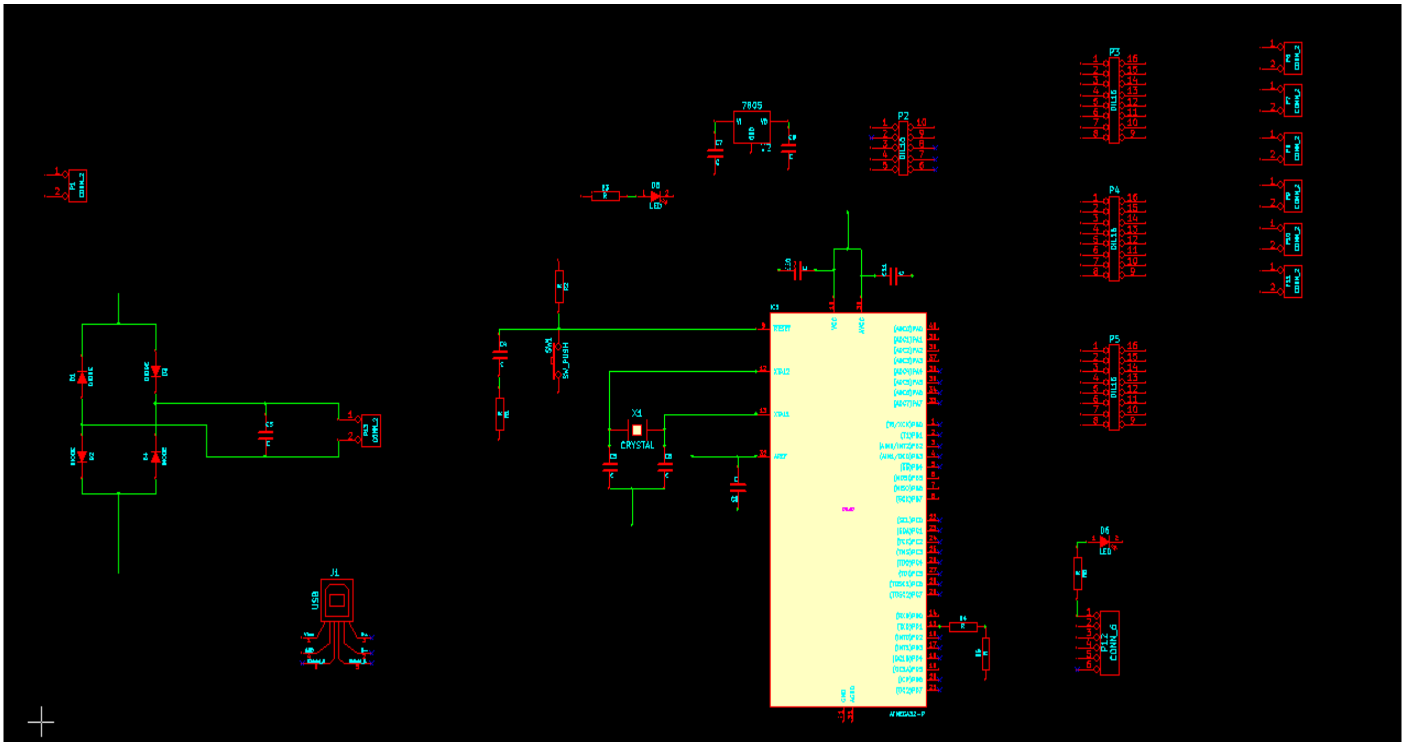
1. First, we tested only the motor driving circuit by putting appropriate codes on Atmega 32A.
2. we then tried to capture live video and track the object through pi camera and do necessary processing over it .

3. Then we tried to pass signal to the atmega32a through raspberry pi .
4. we then tried to smoothen the delay that was causing a serious problem while transferring data and communicating the devices.
5. The overall system was then tested for accuracy on different angles, directions and positions.
6. Finally, we assembled everything and tested the complete prototype several times.

BLOCK DIAGRAM



4.4 CIRCUIT DIAGRAM



4.5 TESTING MEACHANISM

The testing mechanism first includes the capture of image with the help of camera board of Raspberry Pi and detected objects of only desired color and eliminated all other objects of other colors with the help of algorithm programmed using Matlab. We then observed objects that fell under the area as per our requirement. Furthermore, a mean center of the object was calculated and displayed on the window so as to identify the position of the object and the required direction of motor for the rotation of the bot was selected and the value was passed to the AVR. The value was displayed in the Matlab window for the better debugging of the possible error . All of the above testing results were considered in designing the product.

4.6 EVALUATION MECHANISM

We evaluated the system based on the performance of the image tracking mechanism and the change of direction of the bot according to the change in the position of the object. The raspberry Pi and the AVR module were good in performance. Range of connectivity also varied depending on the movement and speed of change of the position of the bot. The range of the Wi-Fi was also strong enough to operate the local machine Raspberry Pi in optimum performance level.

4.7 REQUIRED HARDWARE AND SOFTWARE TOOLS

- **Hardwares**

1) AVR Development Board

AVR 40 Pin ***development Board*** is ideal for learning ***AVR*** microcontrollers. This board contains requirements like LEDs, Switches, in system programming, RS232 communication, etc . All Ports of controller are accessible though standard 10 pin FRC connector or simple header pins. This board help us for the implementation of our program to control the various features of our project like controlling the direction of motors and interfacing with the raspberry Pi for the video streaming mechanism.

2) Usbasp

USBasp is a USB in-circuit programmer for Atmel AVR controllers. It simply consists of an ATMega88 or an ATMega8 and a couple of passive components like resistors, diodes, capacitors, etc. The programmer uses a firmware-only USB driver, no special USB controller is needed which is used to burn the program to the AVR development board.

3) Raspberry PI with cameraboard

The **Raspberry Pi** is a series of small single-board mini computers of credit card size capable of performing various programs that can be used in electronics projects, and for many of the things that our computer does, like spreadsheet, word processing, browsing the internet etc. The Raspberry Pi model we have used is the **Raspberry *Pi 2 Model B*** consisting of Quad-core ARM Cortex-A7 CPU 0.9 GHz.

The **Raspberry Pi Camera Board v2** is a high quality 8 megapixel Sony IMX219 image sensor custom designed add-on board for Raspberry Pi, featuring a fixed focus lens. It's capable of 3280 x 2464 pixel static images, and also supports 1080p30, 720p60, and 640x480p90 video.

It attaches to the Pi by way of one of the small sockets on the board's upper surface and uses the dedicated CSi interface, designed especially for interfacing to cameras.

4) **Digital Multimeter and Continuity Testing**

Continuity is the presence of a complete path for current flow. A circuit is complete when its switch is closed. A digital multimeter's Continuity Test mode can be used to test switches, fuses, electrical connections, conductors and other components. A DMM emits an audible response (a beep) when it detects a complete path. The beep, an audible indicator, permits technicians to focus on testing procedures without looking at the multimeter display.

- **Software**

1) Proteus Design Suite

The Proteus Design Suite is a proprietary software tool suite used primarily for electronic design automation. The software is used mainly by electronic design engineers and technicians to create schematics and electronic prints for manufacturing printed circuit boards. We used this software for initial simulation.

2) Kicad

KiCad is a free software suite for electronic design automation. It facilitates the design of schematics for electronic circuits and their conversion to PCB designs with appropriate design connection and 3D visualization. We have used this software for the pcb design of our AVR development board and the final pcb board required for this project.

3) Matlab

MATLAB (**mat**rix **lab**oratory) is a multi-paradigm numerical computing environment. A proprietary programming language developed by mathwork, MATLAB allows matrix manipulations, plotting of functions and data, implementation of algorithms, creation of user interfaces, and interfacing with programs written in other languages, including C, C++, C#, Java, Fortran and Python. We have used this software for the development of image processing and interfacing mechanism between two serial devices.

5) Atmel Studio

Atmel Studio 7 is the integrated development platform (IDP) for developing and debugging all AVR and SAM microcontroller applications. The Atmel Studio 7 IDP gives us a seamless and easy-to-use environment to write, build and debug our applications written in C/C++ or assembly code. It also connects seamlessly to the debuggers, programmers and development kits that support AVR and SAM devices. We have used this software to generate code for the controlling mechanism of our motor and interfacing mechanism between two serial devices.

11) Results

11.1 Result from the Matlab window

With the detection of the desired object which is a circular ball, the program gives us the information about the position of the object in viewing window by displaying the center of the object. Moreover, it also gives the information about the position of the motion of the motor in accordance to the signal it is transmitting to the AVR that controls the overall moment of the motor required to track the object. When the object is out of the frame of the camera it displays a no circle found line to inform the viewer and rotates the bot in order to detect the next frame to allocate the object.

12) DISCUSSION AND ANALYSIS

The project was really helpful for us to really understand many things related to hardware and their intercommunicating ideas. The project has successfully widened our vision to such a extent that can prove to be helpful for our practical knowledge. The theories studies in our instrumentation and electronics came out of paper into a real field of practicability. The project has importance of its one idea and can surely be enhanced further .

13) PROJECT PLANNING AND GANTT CHART

This project being our instrumentation project, we had to face numerous problems and challenges during the course of its completion. A lot of issues occurred in hardware as well as software design. So, for those who are thinking of doing similar kind of project in future, here are some of the challenges we encountered and their respective resolutions.

- **TIME MANAGEMENT**

During the initial phase of our initiative, we had very little idea about how, where and when to kick start our process. We were initially stuck on to which part of the project we start first and how to divide the work among the team. This lack of solid planning lagged us behind in time.

So, it is of immense importance to have a clear idea about how to properly manage time for all the prospects of your project. For this, a concrete strategy needs to be made on how to appropriately part time for all the work or research or anything that needs to be done for the project. Moreover, a proper timeline of all the tasks should be organized at the very beginning and team members need to cope up with it as much as possible. This helps your project run smoothly.

- **EXPENDITURE MANAGEMENT**

We were a bit ignorant regarding the expenses that we made during our project development. We weren't able to do a thorough preparation concerning the elements that we needed, how much we needed and how could be deduce our expenses. This led our expenditure to be a bit more than it could have been.

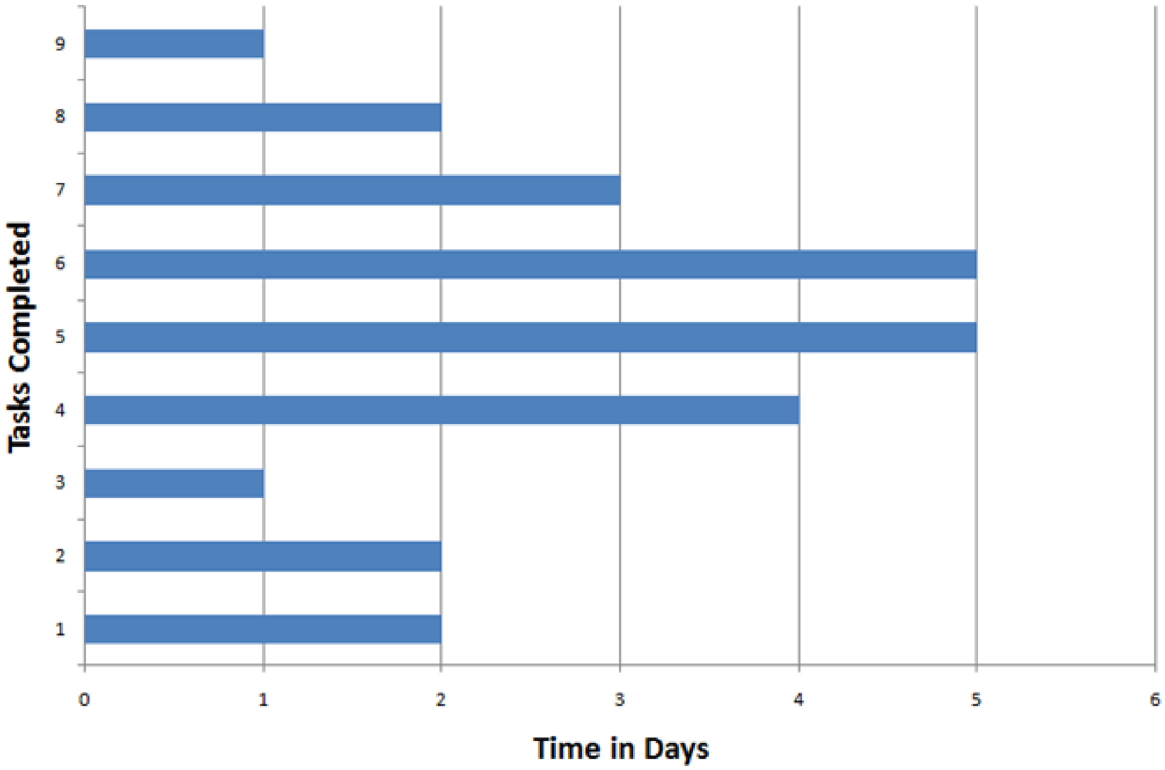
So, we suggest to having a clear planning of how to handle your project budget, where and how to spend it properly.

- **ERRORS IN HARDWARE CONNECTIONS**

It is of huge importance to design perfectly working hardware connections for any electronic project. But it's easier said than done. In our project too, we had to face numerous errors and Undetectable problems during the course of our hardware design. Sometimes, we got wedged for days in a single problem. This really affected our timeline.

Although it's not viable to avoid any kinds of errors, the best way to minimize them would be to accurately understand the devices that are being used. Datasheets of all the components must be thoroughly studied and always try to maintain the standard conditions for them that are mentioned there. Use various CAD software for checking the circuit connections first and then implement them on actual hardware. This saves you time and effort. Also, try to do these things in group of your team members because mistake done by one can be caught by another

GANTT CHART



.

INDEX	TASK COMPLETED
1	Basic research and study
2	Buying the components required
3	bread boarding circuit
4	Hardware Circuit Design
5	Learning AVR Programming and writing AVR codes
6	Testing pcb circuits
7	Setting up Raspberry Pi and interfacing with its camera module
8	Miscellaneous
9	Debugging and final detailing

14) BUDGET ANALYSIS

Due to the uncertainty of expenses that we made during our project development, we weren't able to do a thorough preparation concerning the elements that we needed, how much we needed and how could be deduce our expenses. This led our expenditure to be a bit more than it could have been. However we were provided with various hardware supports by our respected seniors which reduced our expense to greatest possible value.

TOTAL EXPENDITURE

SN	Name of Components	Used No.	Price per each (Rs)	Total (Rs)
1	DC Motor	6	250	1500
2	LIPO battery	1	1000	1000
3	AVR chip(ATmega32A)	1	300	300
5	Motor Driving chip(L293D)	3	50	150
6	Raspberry Pi 3	1	5800	5800
7	Camera Module	1	2500	2500
Grand Total				11,250

15) CONCLUSION

We made a small attempt to represent our project through prototype of a car and used a small green ball to get tracked as object. The project can be enhanced further to follow not only an object but anything since the entire processing is based upon image processing. Only with the alternation of few algorithms and codes we can be able to accomplish anything as big as tracking human itself. The project is only a small demo of what can be achieved further and has opened it scope too.

17) REFERENCES

1. Official Matlab documentation website

   https://www.mathworks.com/

2. https://www.youtube.com/watch?v=T_ekAD7U-wU