DESIGN OF AGV WITH MOBILE COMMUNICATION

[1]M.RUKMINI SAI RUPA SRI, [2]Y.SHIVRAJ NARAYAN, [3]K.LOKENDER

VNR Vignana Jyothi Institute of Technology, VNR Vignana Jyothi Institute of Technology, MVSR Engineering College
Email: mada.rukmini@gmail.com, shivrajyeole@vnrvjiet.in, k.lokender@gmail.com

Abstract- Automatic guided vehicles (AGVs) that run automatically and are controlled by computers have been used in a wide range of fields and applications, such as on factory production floors, in warehouses, and in distribution centres, and have received more attention as a mode of transportation because they do not produce CO_2 emissions that harm the environment. Due to the abundant availability of human labour at lower costs, these are rarely implemented in India. Even then implementing AGVs will bring many benefits to the organization like improved safety, reliability and efficiency. The design of AGV can be quite challenging and the cost and performance of the system are greatly impacted. The main objective of this work is to design and develop a cost effective AGV. This paper presents the design of AGV and its control using ATmega16, along with its fabrication which is a line follower. The robot is reprogrammable and changes can be made so it suits the real industrial environment. It is allowed to follow accurately the magnetic tape path laid along the flat floors. Test runs are conducted by programming for a predesigned path providing it with an imitating industrial environment and also obstacles in its path. PID control is also included in this vehicle for smoother movements along the path.

Keywords: AGV, ATmega16, Line follower

I. INTRODUCTION

An automated guided vehicle (AGV) is a mobile robot that follows markers or wires in the floor, or uses vision or lasers. These are most often used in industrial applications to move materials around a manufacturing facility or a warehouse [1] [5]. These are intelligent material handling systems. Automated guided vehicles increase efficiency and reduce costs by helping to automate a manufacturing facility or warehouse. AGVs are employed in nearly every industry, including, pulp, paper, metals, newspaper, and general manufacturing. Transporting materials such as food, linen or medicine in hospitals is also done [5]. The design of an AGV has to be done to maximize its usability. The basic concept of AGV is a battery-powered, driverless vehicle with programming capabilities to select its path and position at loading or unloading station. The mechanical design concepts combine the knowledge of various mechanical parts such as motors, wheels, its structure and other essential mechanical parts. The electronic concepts include the concepts of microcontroller, voltage regulator, motor drivers and other electronic parts required for the system [6].

The vehicle is designed to trace the path and avoid obstacles automatically using IR sensors [2]. The path laid is a magnetic black tape [3]. The microcontroller used in this vehicle is Atmega16 [4] which store the instructions for the running of the vehicle.

II. DESIGN SPECIFICATIONS

A. Microcontroller

A microcontroller is a small computer on a single integrated circuit containing a processor core, a memory unit and programmable input/output peripherals. They perform actuating function of an automatic guided vehicle. Their purpose is to keep the device on track as per the signals received. These are designed for embedded applications unlike microprocessors used in personal computers or other general purpose applications.

The microcontroller used in AGV is ATmega 16. The reasons for using this microcontroller are:

- Cost effective
- C cross compiler and library is available for programming
- Availability of UART communication- can directly communicate with PC

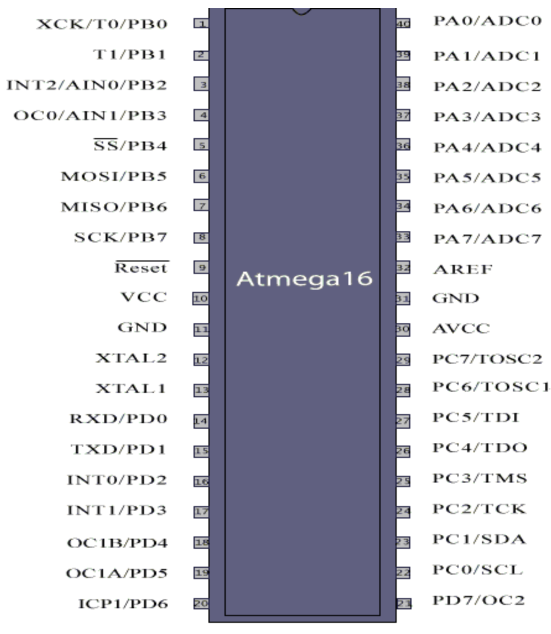


Fig.1 Pin Diagram of ATmega16 Microcontroller

FEATURES:

- Advanced RISC Architecture
- 131 Powerful Instructions

- Most Single Clock Cycle Execution
- 32 x 8 General Purpose Working Registers
- Up to 6 MIPS Throughput at 16MHz
- Fully Static Operation
- Non-volatile Program and Data Memories
- 16k Bytes of In-System Self-Programmable Flash
- 512K Bytes EEPROM
- On-chip Analog Comparator
- Two 8-bit Timer/Counters with Separate Prescaler, Compare
- One 16-bit Timer/Counter with Separate Prescaler, Compare and Capture mode
- Real Time Counter with Separate Oscillator
- Four PWM Channels
- 8-channel, 10-bit ADC
- Byte-oriented Two-wire Serial Interface
- Programmable Serial USART
- Power-on Reset and Programmable Brown-out Detection
- Internal Calibrated RC Oscillator
- External and Internal Interrupt Sources
- 32 Programmable I/O Lines
- 4.5-5.5V for ATmega16
- 0-16 MHz for ATmega16

B. MOTOR DRIVER L293D

It is an electronic integrated circuit which enables a voltage to be applied across a load on either direction. It allows a circuit full control over a standard electric DC motor, i.e., with an H-bridge, a microcontroller, logic chip or remote control can electrically command the motor to go forward, reverse, and coast. The current provided by the MCU is of the order 20mA and that required by the motor is approximately 500mA. Hence the motor cannot be controlled directly by Multipoint Control Unit (MCU) and we need an interface between the MCU and motor.

A double pole double throw relay can generally achieve the same electrical functionality as an H-bridge, but a H-bridge would be preferable when smaller physical size, high speed switching, low driving voltage is needed or when the wearing of mechanical parts is undesirable. The term H-bridge is derived from the typical graphical representation of such circuit, which is built with four transistors as switches.

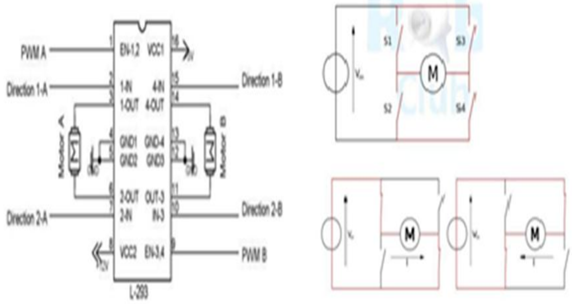
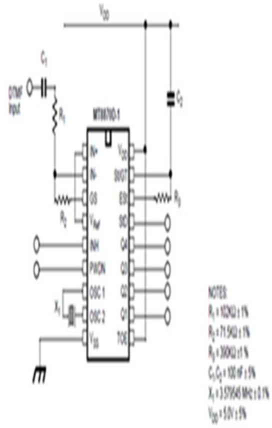


Fig.2 (a) Control Circuit of Motor Driver (b) H – Bridge

C. DTMF DECODER MT8870

The MT8870D/MT8870D-1 monolithic DTMF receiver offers small size, low power consumption and high performance. Its architecture consists of a band split filter section, which separates the high and low group tones, followed by a digital counting section which verifies the frequency and duration of the received tones before passing the corresponding code to the output bus.



F_{LOW}	F_{HIGH}	KEY	TOW	Q_4	Q_3	Q_2	Q_1
697	1209	1	H	0	0	0	1
697	1336	2	H	0	0	1	0
697	1477	3	H	0	0	1	1
770	1209	4	H	0	1	0	0
770	1336	5	H	0	1	0	1
770	1477	6	H	0	1	1	0
852	1209	7	H	0	1	1	1
852	1336	8	H	1	0	0	0
852	1477	9	H	1	0	0	1
941	1209	0	H	1	0	1	0
941	1336	*	H	1	0	1	1
941	1477	#	H	1	1	0	0
697	1633	A	H	1	1	0	1
770	1633	B	H	1	1	1	0
852	1633	C	H	1	1	1	1
941	1633	D	H	0	0	0	0
-	-	ANY	L	Z	Z	Z	Z

L = logic Low, H = Logic High, Z = High Impedance

Fig. 3 (a) DTMF decoder circuit (b) DTMF decoder functional decoding table

D. MECHANICAL AND ELECTRICAL ASPECTS OF DEVICE

For building automatic device a platform type chassis was used. It was equipped with two dc motors, two wheels and sliding sets, in order to increase stability, a rechargeable battery of 12V, electric power for motors and electronic management control, empowered with Atmega16 microcontroller. Platform is provided with an emergency circuit-breaker. LED diodes, infrared sensors and an ultrasonic sensor are mounted at centre bottom of platform. Infrared light emitted by diodes towards white line is reflected to all eight sensors, these send a logical signal to microcontroller using a sensor port. Motor driver control circuit drive motors using Pulse WithModulation (PWM) technique.

III. DRIVE AND CONTROL SYSTEM

The robot was designed as a three wheeler robot with two independently controllable wheels at the back and a free un-powered caster wheel in the front. Such a configuration provided very convenient forward and backward locomotion, as well as steering in any direction just by controlling the speeds and directions of the motors. The caster provided will stabilize the vehicle in turnings. The robot is controlled by ATmega16 micro-controller based system, using feedback signal from numerous sensors. Microcontroller generated PWM signals was utilized to control the drive motor's speeds. Two PWM signals were used to control the speeds of two drive motors of AGV. The motion is planned in such a way that the vehicle slows down if it comes across an obstacle.

Depending on the signals given by sensors either both the motors work in the same direction or one of the motor works while the other motor reverses its motion. When one motor works and the other motor reverses, then this condition causes the AGV to turn in the direction of the motor which has reversed. When both the motors continue to work in forward direction then the AGV will move in a straight path. Thus making use of these conditions the AGV is made to follow the pre-defined path.

IV. LINE DETECTION AND OBSTACLE AVOIDANCE

The system is provided with an array of sensors for path tracing and obstacle detection in the front. 8 pairs of IR receiver and transmitter are used for this purpose. These IR sensors sense the track using the light reflected from the black line on a light background. The sensors then depending on the position send the signals to the Micro-Controller. The signals are processed and depending on the signals received the microcontroller gives the corresponding output to the motor control module such that the AGV continues to follow the path.

According to their detection, the black path is traced along the layout and the vehicle keeps on moving till it reaches its destination as per the communication code. In this way the vehicle continues to follow the pre-defined path which is a black line on a white background.

Ultrasonic sensor is used for obstacle detection purpose. This sensor converts the distance into corresponding analog voltage, which is converted to digital by microcontroller using ADC (Analog to Digital Conversion) conversion. The converted digital value is decoded to distance using a formula. Whenever the vehicle encounters an obstacle along its path, it is programmed to slow down or halt if the obstacle is not removed for a longer duration, to save battery life. A buzzer is provided for emergency indications.

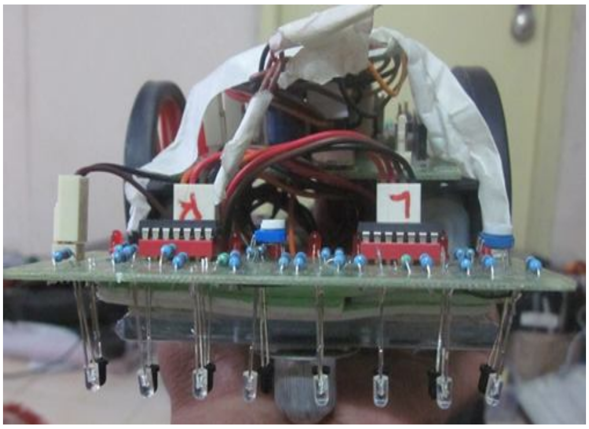


Fig. 4 Eight sensor strip on the vehicle

V. COMMUNICATION BETWEEN USER AND AGV SYSTEM

As we have stated earlier that user (operator) loads the AGV for the stations. It is sent by calling the AGV and has to give input in the following way.

- Input :S1*I1*Q1#S2*I2*Q3##

HereS1 and S2 are the stations where delivery of loads is to be done. I1 and I2represent the item that has to be transferred for their respective stations S1 and S2. Q1 and Q2 represent the quantity of the loads I1 and I2 respectively. '*' symbol separates the station, item name and its quantity, '#' separates inputs and '##' indicates the end of input and the AGV starts once this is typed.

Ex: 1*5*3#2*4*5##

So from the above input the user enters station numbers separated by #, item to be loaded or unloaded, the quantity of the item required each of them separated by * and any number of points can be given in any order. ## is pressed to instruct AGV system to stop receive number from DTMF decoder and all the point received are stored in sorted stack (array).

This raw data is received by the DTMF decoder which is given to the microcontroller to convert into station numbers. As we know each number pressed in the mobile has different frequencies and DTMF decoder differentiates them and sends the corresponding binary codes to the microprocessor and the microprocessor manipulates the input and sends commands to the AGV.

V. PROGRAMMING THE AGV

Most commonly used language for programming microcontrollers is Assembly, but for this design embedded C language is considered, being much faster and easier than Assembly languages.The programming for AVR ATmega microcontrollers is done in Embedded C language as it is simple in understanding and can be used by any user with basic programming skills. Also it easy to develop and maintain in this language whose execution is faster.It provides a single development platform for Atmel's 8-bits, 32-bits and ARM Cortex-M families of AVR microcontrollers. AVR studio 6 is a full software development environment with an editor, simulator, programmer, etc.It comes with its own integrated C compiler, the AVR GNU C Compiler (GCC) eliminating a third party C compiler.

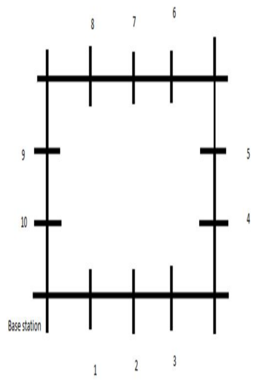
An algorithm is generated to get an explicit idea before starting the programming. Initially, AGV receives the data from the main server. The system compares the received data with the existing data to start its move. If the data matches, with the help of sensors the vehicle traces the path and moves to the respective stations. After finishing its travel for the given code, it moves to the base station and waits there till it receives the next data. If during its travel the next data is received, it stores that data and executes it only after finishing its previous task.

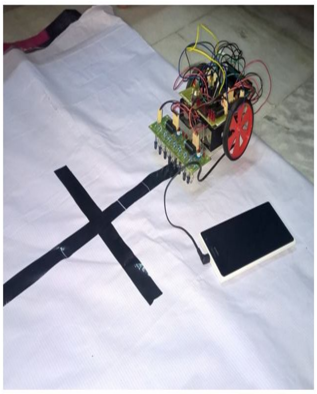
The vehicle follows the above algorithm for its run:

```
main()        // start of main function
{
init_inte();    // interrupt function
InitPorts();    // port calling function
InitPWM();   // Pulse With Modulation Function
While(1)
{
Start AGV();      //function to start AGV
}
}
Call interrupt()     // interrupt calling function when the mobile on the vehicle receives a call from user
{
Take data from DTMF decoder;
Decode the data();
Start AGV();
}
Start AGV()
{
If(stop matches the current station) // comparing the current input with the predefined stations
{
AGV halt() // stops AGV for 60 seconds for loading or unloading the material
}
Else
Run AGV()   // the vehicle moves to the next station if given or to the base station
}
```

VII. EXPERIMENTAL SETUP

An arena based on industrial setup is created on a white sheet and obstacles are placed on it to check the working of robot. The arena has one base station where it is loaded and several delivery stations where the material is unloaded by the users as per their requirement. After completion of its delivery in all the stations as per the formula the bot returns to its base station. The vehicle has intelligence of taking the shortest route to the stations.





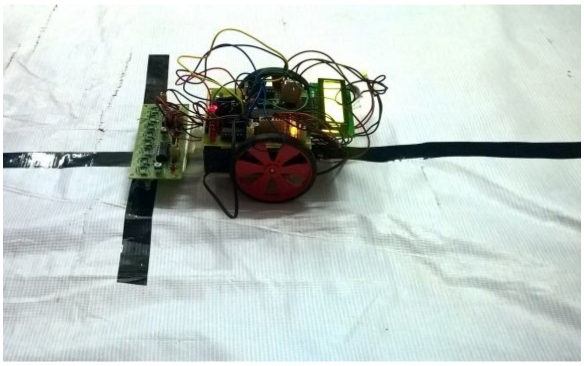


Fig.5 (a) Arena for vehicle movement (b) AGV taking Input using Cell phone (c) Vehicle stopping at a Station

VIII. RESULTS

Test runs are conducted in a dummy industrial setup for checking the proper functioning of the robot. The vehicle is run along the path of the arena i.e., the black line and the response of it is checked with variations in path, obstacles and environment. The sensor array is checked precisely by inspecting the function of each and every sensor in it. For electronic safety, the circuit was checked so the ICs do not cause any problem while running the robot. The wheel system was adjusted for the variations of load that the vehicle has to carry. The motors are also tested for their performance with current fluctuations. Thus, the robot is proved as a configurable robot which is available in low budget and can be dependable.

FUTURE WORK

The vehicle is been tested for the four wheel drive system with an appropriate steering mechanism, to increase the load capacity of the robot. If the vehicle has to be employed in sunlight, replacing the current IR sensorswith TSOP sensors on this system will allow it to move even in sunlight. By integrating Image processing technique with laser navigation or Sonars predefined path can be eliminated which makes AGV completely autonomous.

The current AGV control system can be split into Server-Client system where the server takes responses and instructs the clients which are AGVs to move along a Grid system.To further reduce the manual work, a robotic arm or lifting mechanism can be incorporated on AGV for loading or unloading the material.

REFERENCES

[1] R. Siegwart and I. R. Nourbakhsh,“Introduction to Autonomous Mobile Robots,” 2nd edition (MIT Press, 2004)

[2] Sowndar Paul. J, “An Intelligent AGV with Lane Detection and Obstacle Avoidance Using ARM,” International Journal of Engineering Sciences & Research Technology, 2014, pp. 671-677

[3] K. Kishore Kumar, M. Siva Krishna, D. Ravitej, D. Bhavana, "Design of Automatic Guided Vehicles," International Journal of Mechanical Engineering and Technology (IJMET), Vol. 3,2012, pp. 24-32

[4] Kanis Fatima, Md. Abul Hossain, "Microcontroller ATmega32 Based Automatic Vehicle Control," International Journal of Scientific& Engineering Research, Vol. 5,2014, pp. 1-5

[5] Groover Mikell P, "Automation, Production Systems, and Computer - Integrated Manufacturing," Third Edition, PHI publications,2008

[6] Rone Poumai," Electronics projects," Vol 24, EFY enterprises Pvt. Ltd., pp. 173-174

★★★