International Journal of Computer and Communication Technology

Volume 5 | Issue 4 Article 11

October 2014

# OBSTACLE DETECTION AND NAVIGATION OF AUTONOMOUS ROBOT

HIMANSU SEKHAR SASMAL
*Institute of Technical Education & Research ,Bhubaneswar, Odisha, India*, himansusasmal@gmail.com

TAPAN NAYAK
*Institute of Technical Education & Research ,Bhubaneswar, Odisha, India*, chiku.tapan@gmail.com

Srikanta Patnaik
patnaik_srikanta@yahoo.co.in



Recommended Citation
SASMAL, HIMANSU SEKHAR; NAYAK, TAPAN; and Patnaik, Srikanta (2014) "OBSTACLE DETECTION AND NAVIGATION OF AUTONOMOUS ROBOT," *International Journal of Computer and Communication Technology*: Vol. 5 : Iss. 4 , Article 11.
DOI: 10.47893/IJCCT.2014.1258
Available at: https://www.interscience.in/ijcct/vol5/iss4/11

# OBSTACLE DETECTION AND NAVIGATION OF AUTONOMOUS ROBOT

**HIMANSU SEKHAR SASMAL, TAPAN NAYAK. PROF (DR) SRIKANTA PATNAIK.**

Institute of Technical Education & Research ,Bhubaneswar, Odisha, India

Email: himansusasmal@gmail.com , chiku.tapan@gmail.com , patnaik_srikanta@yahoo.co.in

**Abstract-**This report presents an obstacle detect autonomous mobile robot. Robot used an Atmega 32 microcontroller as its brain. Atmega 32 receives input from Ultrasonic Distance Meter (UDM). Some computations are performed on this input and a control signal is generated to control the robot's position. This control signal is generated through a PWM signal. which is implemented in the microcontroller. Afterwards, microcontroller was programmed in C language using a AVR Studio4 compiler.

***Keywords****: microcontroller, Ultrasonic Sensor, AVR Studio4 compiler, extreem burner, PWM signal*

## INTRODUCTION OF AUTONOMOUS ROBOTS

What is the _rst thing that comes to mind when you think of a robot? For many people it is a machine that imitates a human-like the androids in Star Wars, Terminator and Star Trek: The Next Generation. However much these robots capture our imagination, such robots still only inhabit Science Fiction. People still haven't been able to give a robot enough 'common sense' to reliably interact with a dynamic world. However, Rodney Brooks and his team at MIT Artificial Intelligence Lab are working on creating such humanoid robots. The type of robots that you will encounter most frequently are robots that do work that is too dangerous, boring, onerous, or just plain nasty. Most of the robots in the world are of this type. They can be found in auto, medical, manufacturing and space industries. In fact, there are over a million of these type of robots working for us today. Some robots like the Mars Rover Sojourner and the upcoming Mars Exploration
Rover, or the underwater robot Caribou help us learn about places that are too dangerous for us to go. While other types of robots are just plain fun for kids of all ages. Popular toys such as Teckno, Polly or AIBO ERS-220 seem to hit the store shelves every year around Christmas time. A robot has these essential characteristics:
1. Sensing First of all your robot would have to be able to sense its surroundings.It would do this in ways that are not unsimilar to the way that you sense your surroundings. Giving your robot sensors: light sensors (eyes), touch and pressure sensors (hands), chemical sensors (nose), hearing and sonar sensors (ears), and taste sensors (tongue) will give your robot awareness of its environment. 2. Move-
ment A robot needs to be able to move around its environment. Whether rolling on wheels, walking on legs or propelling by thrusters a robot needs to be able to
move. To count as a robot either the whole robot moves, like the Sojourner or just parts of the robot moves, like the Canada Arm. 3. Energy A robot needs to be able to power itself. A robot might be solar powered, electrically powered, battery powered. The way your robot gets its energy will depend on what your robot needs to do. 4. Intelligence A robot needs some kind of "smarts." This is where programming enters the pictures. A programmer is the person who gives the robot its 'smarts.' The robot will have to have some way to receive the program so that it knows what it is to do.

Well it is a system that contains sensors, control systems, manipulators, power supplies and software all working together to perform a task. Designing, building, programming and testing a robots is a combination of physics, mechanical engineering, electrical engineering, structural engineering, mathematics and computing. In some cases biology, medicine, chemistry might also be involved. A study of robotics means that students are actively engaged with all of these disciplines in a deeply problem-posing problem-solving environment.
1. Find the simplest/minimal set of solutions to solve the most/complex problems this is called _nding an "elegant solution.
2. That is to ask: What is the simplest nervous system an animal can have in order to solve the most complex problems encountered by a simple organism, such as an insect.
3. We choose not to study complex high-level vision, such as scene analysis, because it is overkill to the problem.
4. Instead, we choose to study simple compound eye without even forming a retinal image, yet insects can detect a great variety of visual objects, and escape from predation without even having a brain.
5. Autonomous robots are similar in using the simplest set of algorithms to solve the most complex problems without relying on a super-computer to solve these problems.

## RATIONALE

1. Autonomous robots can mimick the basic functions of simple organisms, such as insects, such that the basic principles of operation used to interact with the environment can be explored.
2. In designing an autonomous robot, we need to address the issues of how an organism, such as an insect, solves the problems encountered in the interaction with the environment.

3. That is to say, the organism needs to "know" or "acquire knowledge" about the environment so that it can respond appropriately in the changing environment in order to survive.

4. Thus, an autonomous robot faces the same issues as an insect would have encountered.

## PROJECT OVERVIEW

Autonomous robots are smart robots that are capable of making decisions to interact with the environment.

## TOOLS, TECHNIQUE AND METHODOLOGY

HardwareTools

Atmega 32 Microcontroller Development board

**Features**

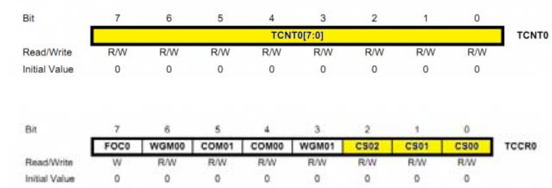
1 The microcontroler is the brain of robo.. Onboard MAX232 interface circuit for easy communication with a computer and other serial devices(GPS modules, GSM Modems, etc).
2. DC plug-in jack for power input.
3. Onboard bridge recti_er enables the board to accept both AC and DC input voltages.
4. Onboard 5V regulator(LM7805) with _ltering capacitors and heatsink. Can accept input voltage in the range 7 - 30V.
5. Power Status LED(Green) and a general purpose LED(Red) connected to PB0.
6. The AVCC pin is connected to 5V through a 100 uH inductor for accurate ADC conversions.
7. Quartz crystal oscillator circuit.
8. Vin, 5V and Gnd bus provided.

## ATMEGA 32 TIMER

as per the following formula, with a clock frequency of 32 kHz and 8-bit counter, the maximum delay possible is of 8 ms. This is quite low (for us, but not for the MCU). Hence for a delay of 6 ms, we need a timer count of 191. This can easily be achieved with an 8-bit counter (MAX = 255).

$$Timer\ Count = \frac{Required\ Delay}{Clock\ Time\ Period} - 1$$

Thus, what we need to do is quite simple. We need to keep a track of the counter value. As soon as it reaches 191, we toggle the LED value and reset the counter. For this, we need the help of the following registers. This is where the uint 8-bit counter of the timer resides. The value of the counter is stored here and increases/decreases automatically. Data can be both read/written from this register. Now we know where the counter value lies. But this register won't be activated unless we activate the timer! Thus we need to set the timer up. How? Read on TCCR0 Register



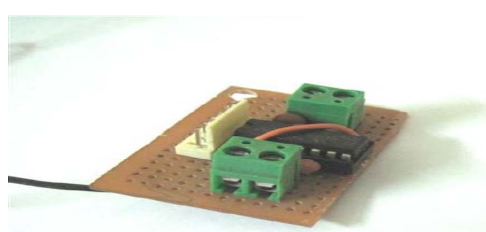

Right now, we will concentrate on the highlighted bits. The other bits will be dis cussed as and when necessary. By selecting these three Clock Select Bits, CS02:00, we set the timer up by choosing proper prescaler. The possible combinations are shown below. TCCR0 Register Right now, we will concentrate on the highlighted bits. The other bits will be discussed as and when necessary. By selecting these three Clock Select Bits, CS02:00, we set the timer up by choosing proper prescaler. The possible combinations are shown below. For this problem statement, we choose

| CS02 | CS01 | CS00 | Description |
|---|---|---|---|
| 0 | 0 | 0 | No clock source (Timer/Counter stopped). |
| 0 | 0 | 1 | $clk_{I/O}$/(No prescaling) |
| 0 | 1 | 0 | $clk_{I/O}$/8 (From prescaler) |
| 0 | 1 | 1 | $clk_{I/O}$/64 (From prescaler) |
| 1 | 0 | 0 | $clk_{I/O}$/256 (From prescaler) |
| 1 | 0 | 1 | $clk_{I/O}$/1024 (From prescaler) |
| 1 | 1 | 0 | External clock source on T0 pin. Clock on falling edge. |
| 1 | 1 | 1 | External clock source on T0 pin. Clock on rising edge. |

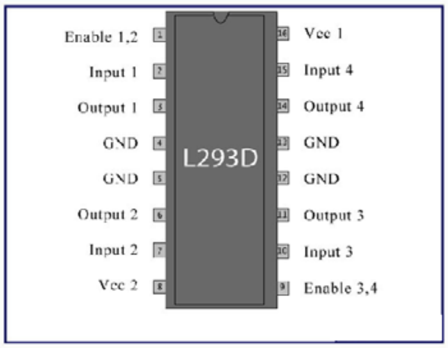
No Prescaling. Ignore the bits highlighted in grey. We will be using it later in this tutorial. Thus, we initialize the counter as: TCCR0 |= (1 << CS00); Please note that if you do not initialize this register, all the bits will remain as zero and the timer/counter will remain stopped. Thus now we are ready to write a code for this. To learn about I/O port operations in AVR, view this. To know about bit Manipulations, view this. Code

I guess the code is pretty simple and straightforward. It doesn't need any explanation. Or maybe one thing needs explanation. In the if statement, I have used if (TCNT0 >= 191) instead of if (TCNT0 == 191) This is because sometimes due to missed compares or unexpected increment, this condition may never be true. Thus to remain on the safer side, we use '>=' instead of ==.

## MOTOR DRIVER(L293D)



L293D is a dual H-bridge motor driver integrated circuit (IC). Motor drivers act as current ampli_ers since they take a low-current control signal and provide a higher-current signal. This higher current signal is used to drive the motors. L293D contains two in built H-bridge driver circuits. In its common mode of operation, two DC motors can be driven simultaneously, both in forward and reverse direction.




The motor operations of two motors can be controlled by input logic at pins 2 and 7 and 10 and 15. Input logic 00 or 11 will stop the corresponding motor. Logic 01 and 10 will rotate it in clockwise and anticlockwise directions, respectively Enable pins 1 and 9 (corresponding to the two motors) must be high for motors to start operating. When an enable input is high, the associated driver gets enabled. As a result, the outputs become active and work in phase with their inputs. Similarly, when the enable input is low, that driver is disabled, and their outputs are o and in the high-impedance state.

## ULTRASONIC DISTANCE MEASUREMENT SENSOR(UD

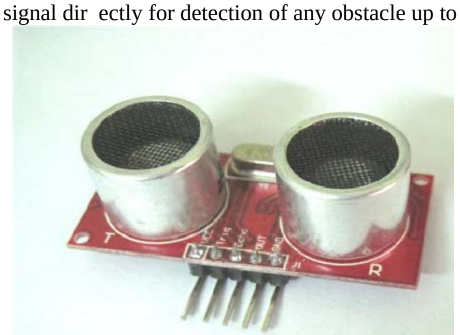
This sensor is a high performance ultrasonic range finder. It is compact and measures an amazingly wide range from 2cm to 4m. This ranger is a perfect for any robotic application, or any other projects requiring accurate ranging information. This sensor can be connected directly to the digital I/O lines of your microcontroller and distance can be measured in time required for traveling of sound signal using simple formula as below.

Distance = (Echo pulse width high time * Sound Velocity (340M/S)/2)

Or

Distance in cm = (Echo pulse width high time (in us)*0.017)

The module works on 5VDC input and also gives an output signal dir ectly for detection of any obstacle up to 4M.



## INFRARED SENSOR

In this project we build an infrared proximity sensor. I used an infrared LED as transmitter and IR TV-like receiver module. Both parts are available at RadioSchack. The IR waves emitted by the LED reect from an object and are caught by the receiver. The IR LED emits 940nm wavelength, has radiant power output 16mW at 100ma (max 1.2A), forward voltage 1.2V, and viewing angle to 1/2 intensity 45. The IR receiver module has passband 950nm50nm and supply voltage and current of 2.4-5.5V and 0.6ma (under no signal output), respectively. The IR receiver module is actually a pretty complicated device. It has a maximum sensitivity not to a steady IR signal, but to a one modulated by 38KHz. It contains _lters to _lter out 38KHz carrier and considers everything else as a noise and ignores those signals. The reception range is rated by 46.2' (approx. 13m). However, this is the case when the module receives a direct waveform transmitted by a remote control unit. In our application the module receives a reflected

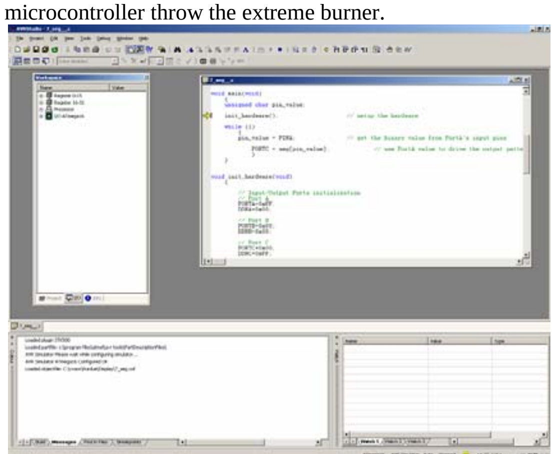
waveform which reduces its range to about a feet. To minimize errors in signal detection the voltage powering the module must be very good _ltered out. The transistor along with the cap and resistor on the schematic do the job.

## SOFTWARE TOOLS

Software required

1. Avr studio4
2. Extreme burner

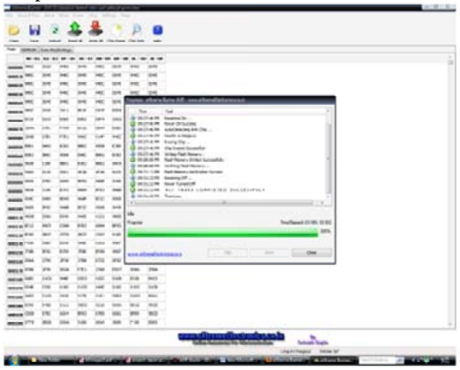
The avr studio4 is required to write atmega 32 chip programs and built the hexcode for atmega32. After the HEX code is generated the this HEX code are written to the microcontroller throw the extreme burner.



Avr studio IDE

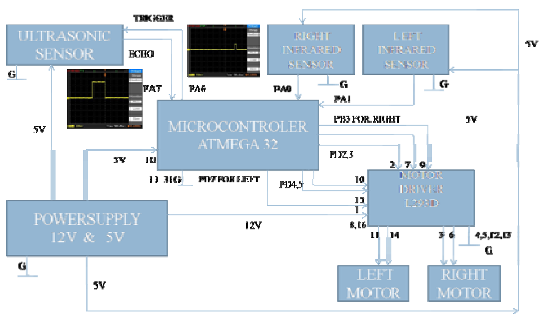
Extreme burner Chip burning tool

The extreem burner is software through which we write the program to flash memory of atmega 32 .Extreem burner download from website. And save it in drive then run or setup the extreme burner.



EXTREEM BURNER IDE

**Develop of Autonomous of Mobile Robot Platform**



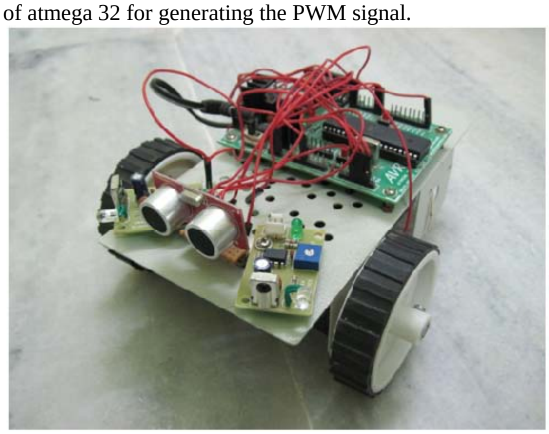

Block diagram

Microcontroller send the trigger signal to ultrasonic sensor trigger pin then out echo signal from sensor .The signal is a pwm signal, the pwm signal vary according front distance. then microcontroller calculate the distance and compare the distance with the threshold distance which is fixed by the programmer . If the calculate distance is less than and equal to thresh hold distance then the robocar move right. If right infrared detect any obstacle the robocar move to left.

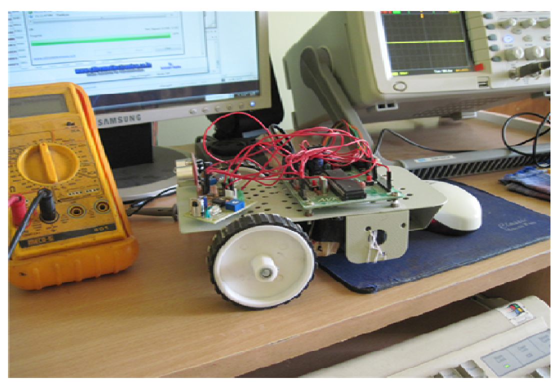
## ASSEMBLING THE HARDWARE COMPONENT

Assembling the hardware using the block diagram.
connection theory
pA0=right infraread signal.
PA1=left infraread signal.
PA6 = ultrasonic sensor triger pin.
PA7= ultrasonic sensor echo pin.
PB3=PWM signal out for right motor speed control , it connect to L293D pin 1.
PD2,3= signal for right motor on o_,it connect to pin 2,7 respectively of L293D.
PD4,5= signal for left motor on o_,it connect to pin 10,15 respectively of L293D.
PD7=PWM signal out for left motor speed control ,and it connect to L293D pin 9.

We fixed two motor and two wheel in side of chassis for movement.
Fixed Development board on top of the chassis.
Fixed caster wheel backside of chassis.
Fixed ultrasonic sensor in front of chassis ,for sending and receiving of ultrasound.
Fixed two infrared sensor in front of chassis one in left and another in right. For detect the obstacle.
Infrared sensor has three pin, pin 1 is 5volt supply,pin2 is ground, pin 3 is signal out. That signal goes to the microcontroller .fixed the motor driver board in side of chassis, for controlling the motor. Using timer0 and timer2 of atmega 32 for generating the PWM signal.



Development board ,motor driver and all sensor uses the power from 12volt battery pack .
The left motor , right motor and caster while attach with chassis . the development board on the top of chassis. Attach the ultra sonic sensor in front of robocar . Fit two infrared sensor in front of robocar in left and right . then connect the ware to main board .



## DISCUSSION AND CONCLUSION

**The Challenge**

1. Design a robot without any a priori knowledge of the environment.
2. That is, the robot has no knowledge of the external environment, and there is no pre-programming done to allow the robot to know beforehand what to expect or how to respond in a given solution.
3. The robot has to learn from the exploratory experience to acquire knowledge about how to respond appropriately in the environment in order to survive.

4. If we pre-program the solution of how to solve the problem, we would be actually cheating, and defect the whole purpose of studying the robot's behaviors.
5. So the question becomes: What is the minimal set of assumptions and constraints that we need in order for the robot to acquire knowledge from the exploration in the environment such that the internal circuitry will be self-organized to produce the final appropriate response similar to those we found in the reflexes in insects.

## THE SOLUTIONS

1. Neural network algorithm is used to "learn" from experience. Exploration by the robot provides the exemplary set of data to build an internal map of the en-vironment.
2. Neural nets are known to adapt to the environment, and learn from the environment without any pre-programming. The task of the neural net essentially construct the nonlinear mathematical mapping functions from the input set (external environment) to the output set (motor responses), with the internal synaptic weights stored as the internal map.

## REFERENCES


[1] Robotic Swarms Project, http://embeddedlab.csuohio.edu/RoboticSwarms/, 2007
[2] Simple Wall Follower Robot and Code, http://www.cse.uta.edu/Robots
[3] R. Braunstingl, J. Mujika, J. P. Uribe, A wall following robot with a fuzzylogic controller optimized by a genetic algorithm, Fuzzy Systems 1995, vol. 5,pp 77-82, March 1995.
[4] Mehmet Ergezer, Multivariable Control Methods for Wall Tracking Robot,Cleveland State University, Department of Electrical and Computer Engi-neering, Project Paper, 2006
[5] Vamsi Mohan Peri, Fuzzy Logic Controller for an Autonomous Robot, Cleve-land State University, epartment of Electrical and Computer Engineering,Masters Thesis, 2005.


❖❖❖