MODELING OF A VEHICLE SPEED MEASUREMENT SYSTEM USING PIEZOELECTRIC SENSORS

A PROJECT THESIS SUBMITTED IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF TECHNOLOGY
IN
ELECTRONICS AND INSTRUMENTATION ENGINEERING

BY
DIBYARANJAN PRADHAN
(107EI007)
RAMAKANT BEHERA
(107EI032)

DEPARTMENT OF ELECTRONICS COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA

2011

MODELING OF A VEHICLE SPEED MEASUREMENT SYSTEM USING PIEZOELECTRIC SENSORS

A PROJECT THESIS SUBMITTED IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

BACHELOR OF TECHNOLOGY
IN
ELECTRONICS AND INSTRUMENTATION ENGINEERING

BY

DIBYARANJAN PRADHAN
(107EI007)
RAMAKANT BEHERA
(107EI032)

UNDER THE GUIDANCE OF
PROF. UMESH CHANDRA PATI

DEPARTMENT OF ELECTRONICS COMMUNICATION ENGINEERING

NATIONAL INSTITUTE OF TECHNOLOGY

ROURKELA

2011

National Institute of Technology
Rourkela

CERTIFICATE

This is to certify that the thesis entitled **"Modeling of a Vehicle Speed Measurement system using Piezoelectric Sensors"** submitted by **Dibyaranjan Pradhan** and **Ramakant Behera** in partial fulfillments for the requirements for the award of Bachelor in Technology degree in Electronics and Instrumentation Engineering at National Institute of Technology, Rourkela (Deemed University) is an authentic work carried out by them under my supervision and guidance.

To the best of my Knowledge, the matter embodied in the thesis has not been submitted to any other University/Institute for the award of any degree or diploma.

Place:Rourkela
Date:16/05/2011

Dr. Umesh Chandra Pati, Ph.D.
Associate Professor
Dept. of Electronics & Communication Engg.
National Institute of Technology
Rourkela - 769008

Acknowledgements

We are heartily thankful to our supervisor, Professor Umesh Chandra Pati, whose encouragement, guidance and support from the initial to the final level enabled us to develop an understanding of the subject. We owe our deepest gratitude to CoEP astronomy club from whom we got the idea to model this system and also to all the staffs of electronics and communication department for their wonderful help and support during our lab access.

Lastly, we would like to offer our regards and blessings to all of those who supported us in any respect during the completion of the project.

Dibyaranjan Pradhan

(107EI007)

Place: Rourkela

Date: 16/05/2011

Ramakant Behera

(107EI032)

Place: Rourkela

Date: 16/05/2011

Table of Contents

Abstract

With increasing number of population and higher rate of development the problem of road accident is also increasing rapidly. So the basic concept is to develop a model that can be useful as a security system in the society and can control vehicle speed.

The property of piezoelectric effect can be used to design a model that can be put on the road surface and whenever a vehicle with higher velocity crosses the sensor it will generate a alarming condition. Two rows or arrays of piezoelectric crystals will be put on the road surface at a fixed distance. Whenever a vehicle passes over the surface of 1st array of sensor it will give a voltage output that can be useful to start the timer in microcontroller and while the vehicle will pass over the second array of sensor, the voltage output from it will deactivate the timer in the microcontroller. Hence the total time taken by vehicle can be determine and so as velocity of vehicle as we know the fixed distance between the two arrays of sensor.

List of Figures

List of tables

Chapter 1

Introduction

Introduction

India is a developing country and it’s developing in a healthy rate. In this rate two decimal GDP is not far away from us. The other thing is India is the second largest populated country in the world. Larger population and healthy economy gives a lot of scope for foreign investment. The core part of this development is increase in earning, better accommodation, better communication and transportation.

As the rate of higher earning people increases in a healthy rate, the number of vehicle running on the roads also increasing in a exponential rate. A lot of vehicle companies are competing in this field. The quality of roads is also increasing which in turn is the cause of higher velocity of vehicles. But the dark side of this development is the increasing rate of road accidents. More than 1,30000 people die in road accidents in a year in India. According to **WHO** (world health organization) India is the no1 country in road accidents. So it becomes extremely important for the government to control the accident rate as well as maintain the development. And the question is how to control it?

In many European country and other developed countries, they are using Laser and Radar based guns to measure vehicle speed and those are based on the principle of measuring the Doppler shift of reflected radiation. But the major problem to accept these technologies in India is its higher cost. Radar based guns starts with a base price of 250$ while laser based gun starts from the base price of 2000$. And when this technology is imported its cost goes much more higher due to different taxation and other overhead.

So the thing is how to measure the vehicle speed which will be cost effective and which will give exact velocity as other equipment like laser guns. India is a very vast country having more than lakhs of road including highways and other important roads connecting different cities within the country. So it is physically impossible to buy such a number of laser or radar guns and also to use them a human operator need a good training of it.

The idea was to develop a robust and inexpensive speed sensor that can sense the speed of vehicle for use in India as well in abroad. The cost of the system should also be taken into consideration while designing the sensor.

The concept of piezoelectric effect can be implemented to have the exact function described above. So a system can be design which will use the concept of piezoelectric effect and can be able to detect the speed of vehicle passes over it. And also it have a buzzer connecting to it so that it can indicate which vehicle speed is more than our desired speed. The basic concept is to use two arrays of piezoelectric crystals on the surface of road at a specific distance. It is possible to calculate the time taken by the vehicle to run from one array of sensors to another. And hence we can calculate its speed as distance between them is known to us.

<u>Chapter 2</u>

Project description

PROJECT DESCRIPTION:

The basic idea is to put two strips of piezoelectric sensors on the surface of road at a fixed distance. The distance between them can be set to any desired value.

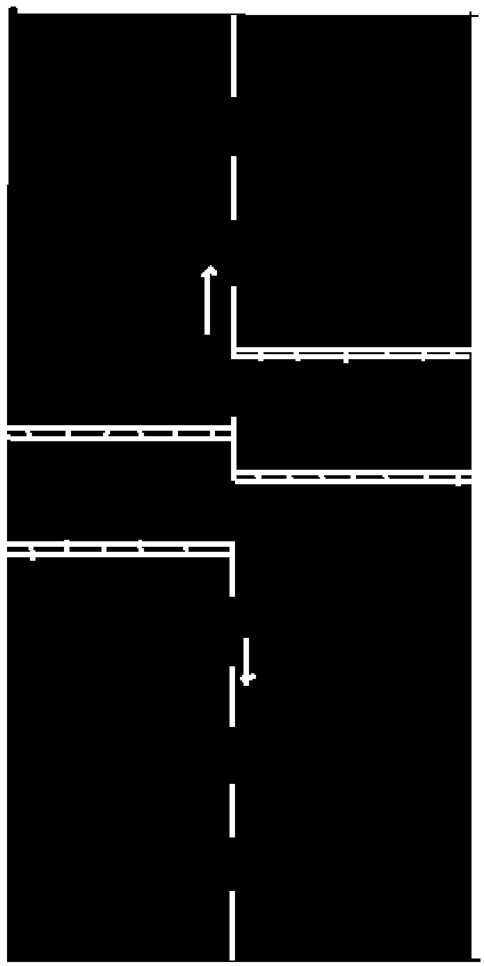

***Figure 2.1**: implementation of sensors on road*

When a vehicle passes over the 1st array of piezoelectric sensor or 1st strip a voltage generates. When the vehicle passes over the 2nd array or strip of piezoelectric sensor another pulse generate due to piezoelectric effect.

These voltages can be used further to activate and deactivate the timer bit in the microcontroller. Hence we can find out what is the exact time taken by the vehicle to cross from one array of sensors to another. Now the velocity can be found out by doing some mathematics in the microcontroller its self. As velocity V= D/T

Where D= distance between the two arrays.

T= time taken by the vehicle to cross the arrays.

If the observed velocity will be greater than a specific velocity then a alarming condition will be generate which will activate a buzzer. And hence we can be sure that the velocity of crossed vehicle is more than the desired speed. Hence this model can be implemented as a security system to control speed of vehicles in the society.

COMPLETE BLOCK DIAGRAM OF THE SYSTEM

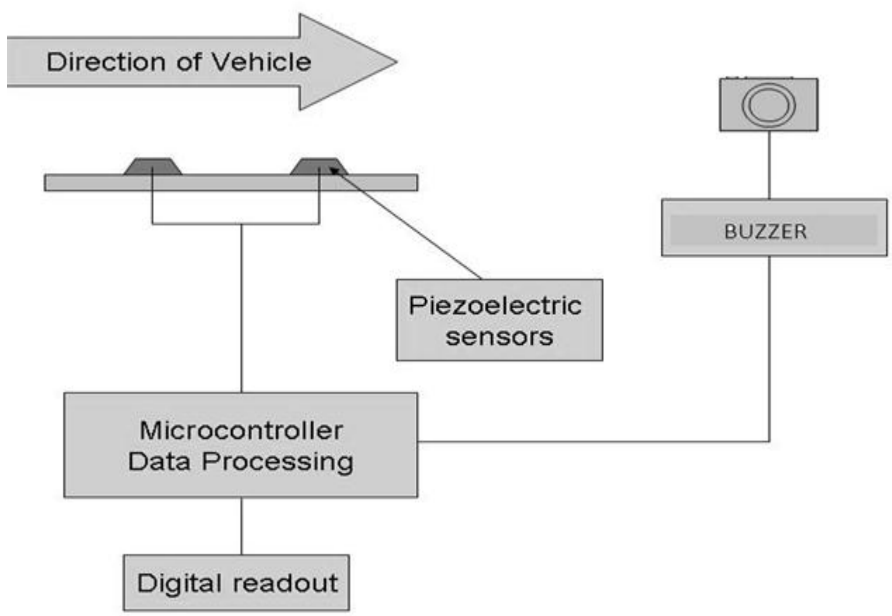

Figure 2.2: complete block diagram of the model [7]

2.1 Problems Encountered in using Piezoelectric Sensor.

- High sensitivity is always a problem with the piezoelectric sensors. The voltage always varies a lot. While holding it on the hand or by simply touching it, it will give a random voltage output of 50 to 80 mv. Its always hard to ignore this random, small but significant voltage output.
- The other thing is while pressing a piezoelectric crystal it will never give a constant voltage. Each time you press it you will definitely get a different voltage output.
- Dynamic response of the sensor varies a lot.
- Physical sustainability. When heavy vehicle will pass on it, it may not sustain with this amount of high pressure.

Chapter 3

An introduction to piezoelectric effect

An introduction to piezoelectric effect

some materials have the property to produce electricity when they are subjected to mechanical stress. This effect can be termed as **piezoelectric effect**. The stress may be in the form of pressing, hitting, or twisting the material to deform it temporarily without causing permanent deformation.

Similarly when these crystals are exposed to electric field then physical deformation can be noticed. The crystal may lengthen or shorten according to the polarity of the field. This effect can be termed as **Inverse piezoelectric effect.**

This effect can be achieved in crystals that have no center of symmetry.. Each molecule has a different polarization, one end is more negatively charged and the other end is positively charged, and is termed as a dipole. The imaginary line that runs through the center of both charges on the molecule is termed as **polar axis**. Polar axes of all of the dipoles directed in one direction in a mono crystal. The crystal is said to be symmetrical because if the crystal can be cut at any point, the resultant polar axes of the two pieces would definitely lie in the same direction as the original. In a polycrystal, there are different areas within the material that have a different polar axis. The poly crystal is asymmetrical because there is no point at which the crystal could be cut that would leave the two remaining pieces with the same resultant polar axis.

Figure 1 illustrates this concept.

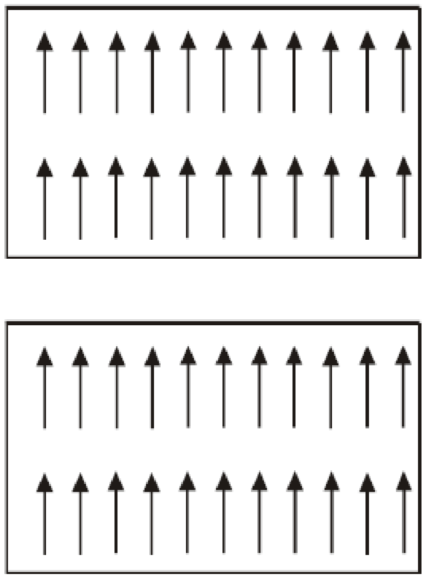

Figure 3.1 mono crystal with single crystal axis [3]

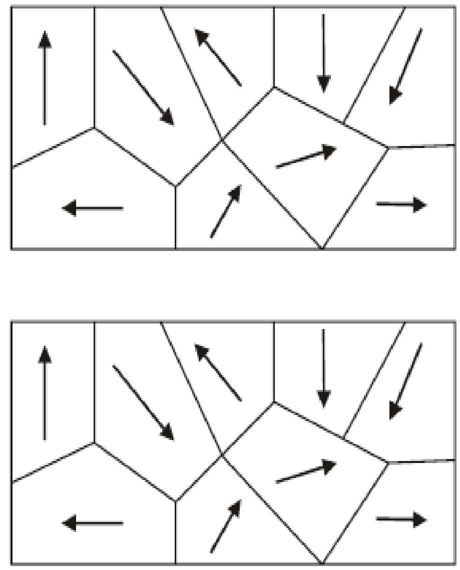

figure 3.2 poly crystal with random crystal axis [3]

To produce piezoelectric effect can be obtain by heating the poly crystal under the application of strong electric field. The heat allows the molecules to move more freely and the electric field forces all of the dipoles in the crystal to line up and face in nearly the same direction.

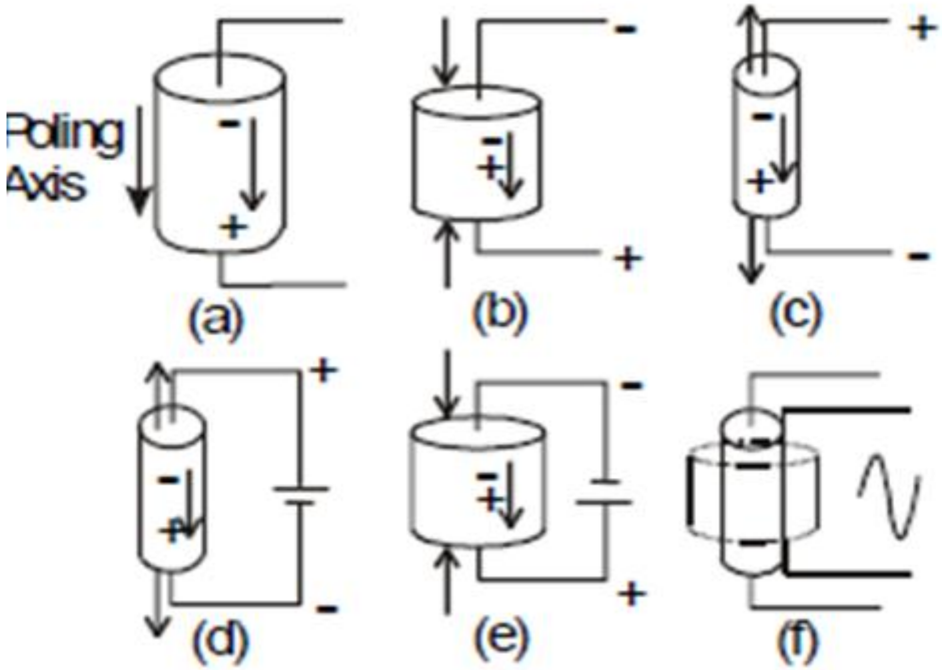

Figure 3.3: piezoelectric and inverse piezoelectric effect [3]

3.1 Different vibration modes in piezoelectric material

The piezoelectric crystal vibrates (bends) differently at different frequency which in other word can be term as vibration mode. Different vibration modes can be achieved by making crystals of various shapes. To realize cost effective and high performance products, several modes have been developed to operate over various frequency ranges. These modes allow us to make products working in the low kHz range up to the MHz range. Different vibration modes and frequency is shown on fig 3.4

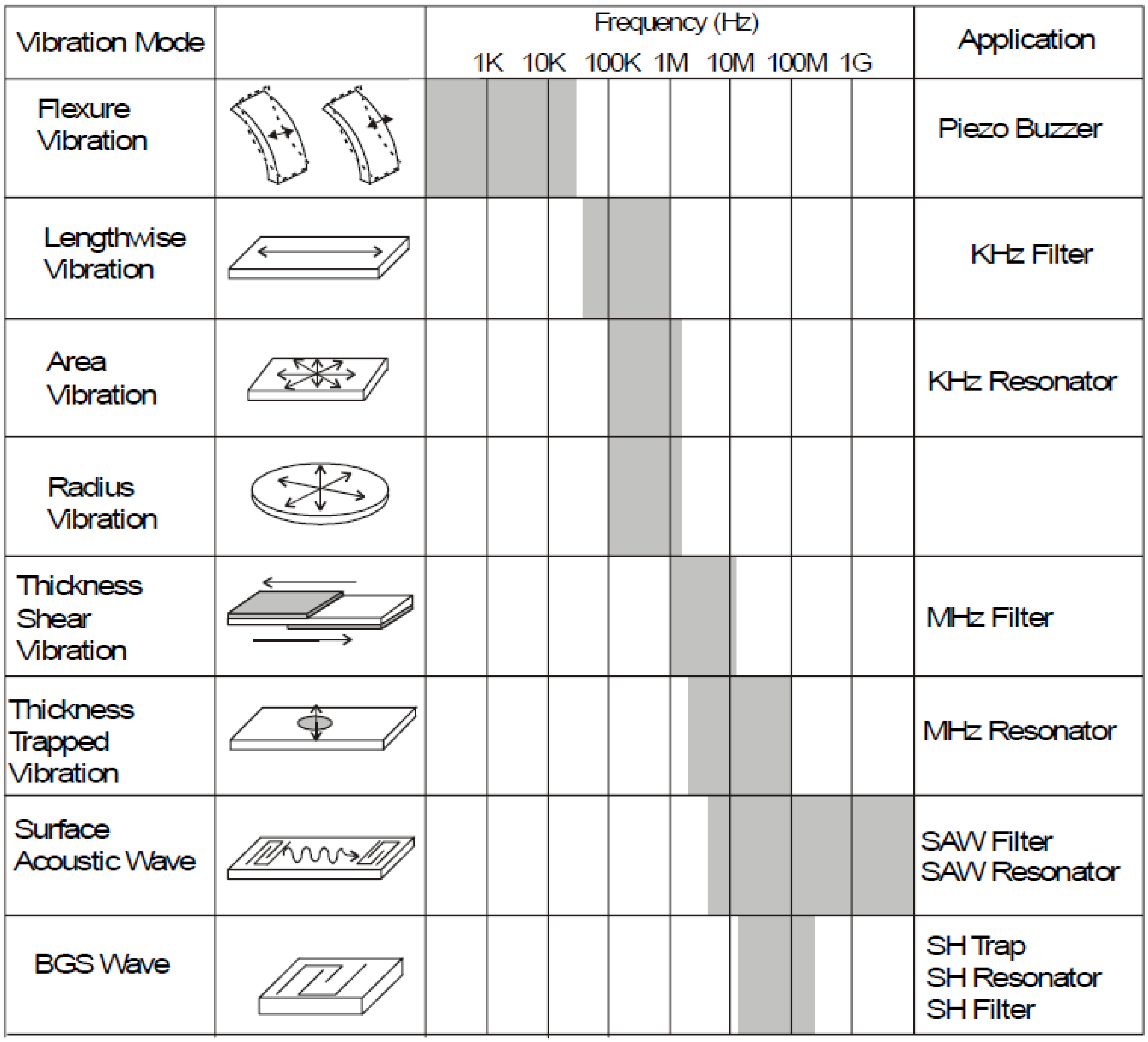

Figure 3.4: Different types of vibration modes of piezoelectric materials [3]

3.2 Types of piezoelectric crystals:

1. Naturally occurring crystals

   - Berlinite
   - Cane sugar
   - Quartz
   - Rochelle salt
   - Topaz

2. Man-made crystals:

   - Gallium orthophosphate
   - Langasite

3. Man-made ceramics:

   - Barium titanate
   - Lead titanate
   - Lithium niobate
   - Sodium tungstate

Chapter 4

Designing of the model

Designing of the model

4.1: Components required:

- **Piezoelectric sensors:** For the modeling purpose we need two piezoelectric sensors. On one, we will give pressure manually which will be responsible to turn on the timer and then the second sensor output will deactivate the timer.
- **Bred board :** it's the one on which we will design a comparator circuit to get a stable voltage output.
- **2 Registers :** one is **2.2 k** and the other one is **1.1 k.**
- **One diode:** a diode is also required.
- **A opamp:** to design a comparator circuit.
- **Atmega32 microcontroller:** a microcontroller is required to on/off the timer in it and also to do some mathematical calculation.
- **LEDs**
- **Digital multi meter :** to check the voltage output
- **Power supply:** For biasing the opamp (+/- 6v)
- **Wires**
- **Microcontroller port connector**
- **Avr studio software :** here a program can be debugged and hex file can be generated.
- **Extreme burner AVR software:** for microcontroller burning
- **+5V Fixed-Voltage Regulator 7805 :** It's a voltage regulator which gives a constant voltage output of 5v whatever may be the variation in input voltage

.

4.2: Electrical characteristics OF L7805

The voltage output will always be 5v whatever variation may the input suffer.

There are other L7800 series voltage regulators are also available for different voltage and current output.

Table 4.1 Electrical characteristics of L7805

Symbol	parameter	Test cond.	minimum	Exact value	maximum	unit
V0	Op voltage	T=25 c	4.8	5	5.2	volt

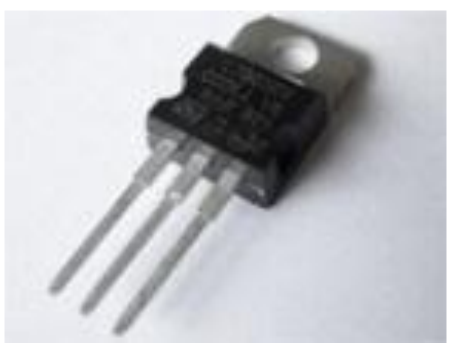

Figure4.1: L7805

4.3: Comparator circuit

It's always difficult to get a constant voltage from the piezoelectric sensors. The voltage output is normally in the range of volt but the current output is generally in Nano ampere range. Each time the sensor is pressed the sensor generates a voltage of 2v to 6v randomly. But we need a constant voltage output. So the voltage output from the sensor is needed to be amplify to get a saturated and constant voltage output.

Again the normal voltage output of the sensor while touching it (not applying force) is also 50-80 mv. If we directly amplify the voltage output from the sensor then in idle condition when no vehicle will pass over it, then also we will get a saturated voltage output. So what we need is a comparator circuit to get a saturated voltage output of around 5v when only the vehicle will pass over the sensor or else when we will press the sensor and otherwise the voltage output will be zero.

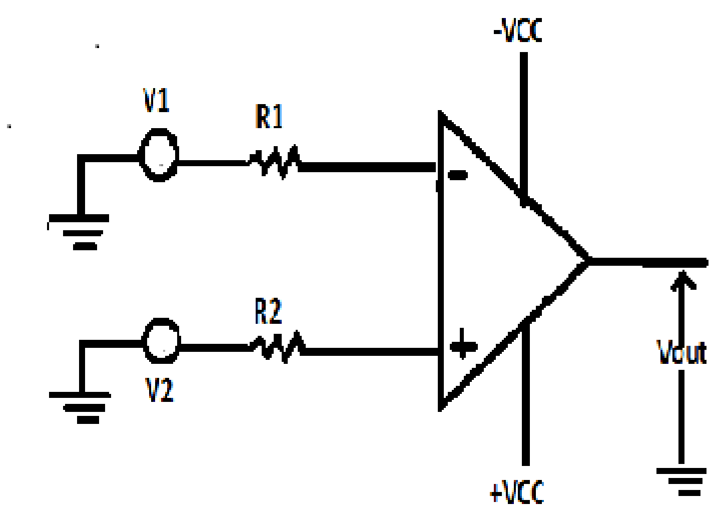

Figure 4.2: comparator circuit

V1=+1.8 v

R1= 1k

R2=2.2k

V2=voltage output from piezoelectric sensor

+/-Vcc =+/-6V

V1 is applied to inverting input of opamp which is pin2.

V2 is applied to non-inverting input of opamp which is pin3.

+Vcc = pin7

-Vcc = pin4

Output is taken from pin6.

Table 4.2: Result of comparator circuit

SRL NO	**PRESSURE APPLIED**	**OUTPUT OF OPAMP**
1	NO PRESSURE	-5.3 V
2	NO PRESSURE	-5.3V

3	1ST PRESS	5.3V
4	2ND PRESS	5.2V
5	3RD PRESS	5.3V
6	4TH PRESS	5.3V
7	5th PRESS	5.3V
8	6th PRESS	5.3V

When V1>V2 that is when no pressure is applied to the sensor then at that time V2-V1< 0

Therefore we got the negative saturated voltage. But as soon as we apply pressure to the sensor then V2-V1 > 0. That's why we got the positive saturated voltage.

But we don't need the negative saturated voltage in idle case rather we need zero volt output in idle case. So the circuit can be modified a little to give the desired output. A diode can be connected at the output of Opamp so that we will get a zero output.

The circuit will be like…

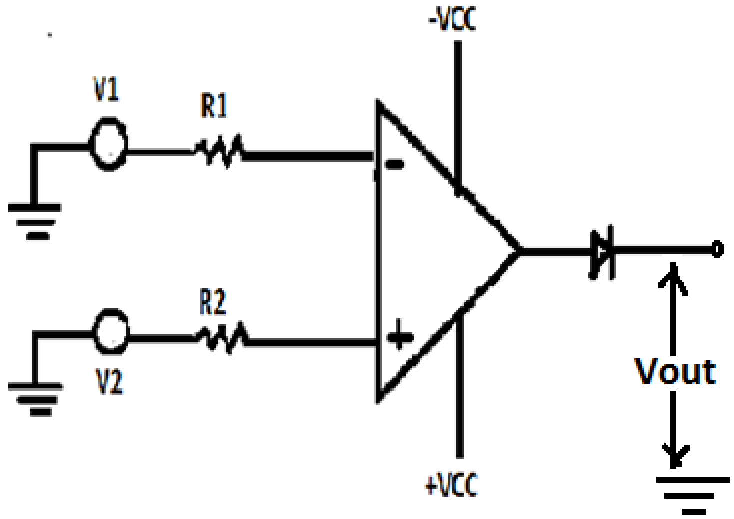

Figure 4.3: comparator circuit with diode

Table 4.3: Result of comparator circuit with diode

SL NO	PRESSURE	OPAMP OUTPUT	DIODE OUTPUT
1	NO PRESSURE	-5.3	ZERO
2	NO PRESSURE	-5.3	ZERO
3	NO PRESSURE	-5.2	ZERO
4	1^{ST} PRESS	5.3V	4.7V
5	2^{ND} PRESS	5.3V	4.8V
6	3^{RD} PRESS	5.3V	4.8V

7	4^{TH} PRESS	5.3V	4.7V

From the given table it is clear that by using a diode we can eliminate unwanted negative voltage output. So when the vehicle pass over the array or when we press the sensor the output voltage will be around 5v. and when there is no vehicle pass activity the comparator voltage output will be zero.

So this voltage can be used to start and stop the timer bit in ATMEGA32 microcontroller.

When the vehicle pass over the first array of sensors it will give a voltage of near 5v to timers input port then the timer will start and when it will move over the second strip then the voltage generated will deactivate the timer. Hence the total time taken by the vehicle to cross the distance between the two strip can be determine.

Chapter 5

Microcontrollers

Microcontrollers

A **microcontroller** is a small computer on a single integrated circuit containing a processor core, memory, and programmable input/output peripherals.

5.1 ATmega32L:

The Atmel AVR Atmega32 is a low power CMOS 8-bit microcontroller based on the AVR enhanced RISC architecture. It combines 32KB of programmable flash memory, 2KB SRAM, 1KB EEPROM, an 8 channel 10 bit A/D Converter and a JTAG interface for on chip debugging. By executing powerful instructions in a single clock cycle, it can achieve throughputs approaching 1MIPS per MHz balancing processing speed and power consumption. Some of the special features are listed below.

1. High performance and low power consumption are the most appealing characteristics of the ATmega32 microcontroller.
2. Advanced RISC architecture is another useful characteristic which comprise of 131 powerful instructions, most of which are executed in a single clock cycle. Its operation is fully static. It has 32x8 general purpose registers and on-chip 2-cycle multiplier. It attains up to 16 MIPS throughput at 16 MHz which is 10 times faster than the conventional SISC architecture based microcontrollers.
3. Memory segment has nonvolatile memory and has high endurance. It includes 32Kbytes of In-System Self-programmable Flash memory, 1024Bytes of EEPROM and 2Kbytes of

Internal SRAM. Write/Erase Cycles for flash memory is 10,000 and that of EEPROM is 100,000. It has Optional Boot Code Section with Independent Lock Bits.

4. This microcontroller also has a JTAG (IEEE std. 1149.1 Compliant) Interface which has capabilities of Boundary-scan according to the JTAG Standard and Extensive on-chip debug support. Flash, EEPROM, Fuses, and Lock Bits can be programmed through this JTAG interface.
5. The peripheral features are excellent. It includes Two 8-bit Timer/Counters and one 16-bit Timer/Counter, all of which has separate prescalers and compare modes. One additional feature of the 16-bit timer/counter is it's capture mode. It also has Real Time Counter with Separate Oscillator and four PWM channels. Its peripheral also include 8-channel, 10-bit ADC, byte-oriented two-wire Serial Interface, programmable serial USART, Master/Slave SPI Serial Interface, programmable watchdog Timer with separate on-chip oscillator and an on-chip analog comparator.
6. Some special Microcontroller features are Power-on Reset and programmable Brown-out detection, internal calibrated RC oscillator, external and internal interrupt sources etc. it has six Sleep modes which are Idle, ADC Noise Reduction, Power-save, Power-down, Standby and Extended Standby. In the idle mode CPU is stopped but it allows the USART, Two-wire interface, A/D Converter, SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning.
7. Its I/O port has 32 Programmable I/O Lines.
8. Power Consumption at 1MHz, 3V, 25^{o}C is 1.1mA in active mode, 0.35mA in idle mode and less than 1µA in power-down mode.

5.2 Pin Configuration of ATmega32L:

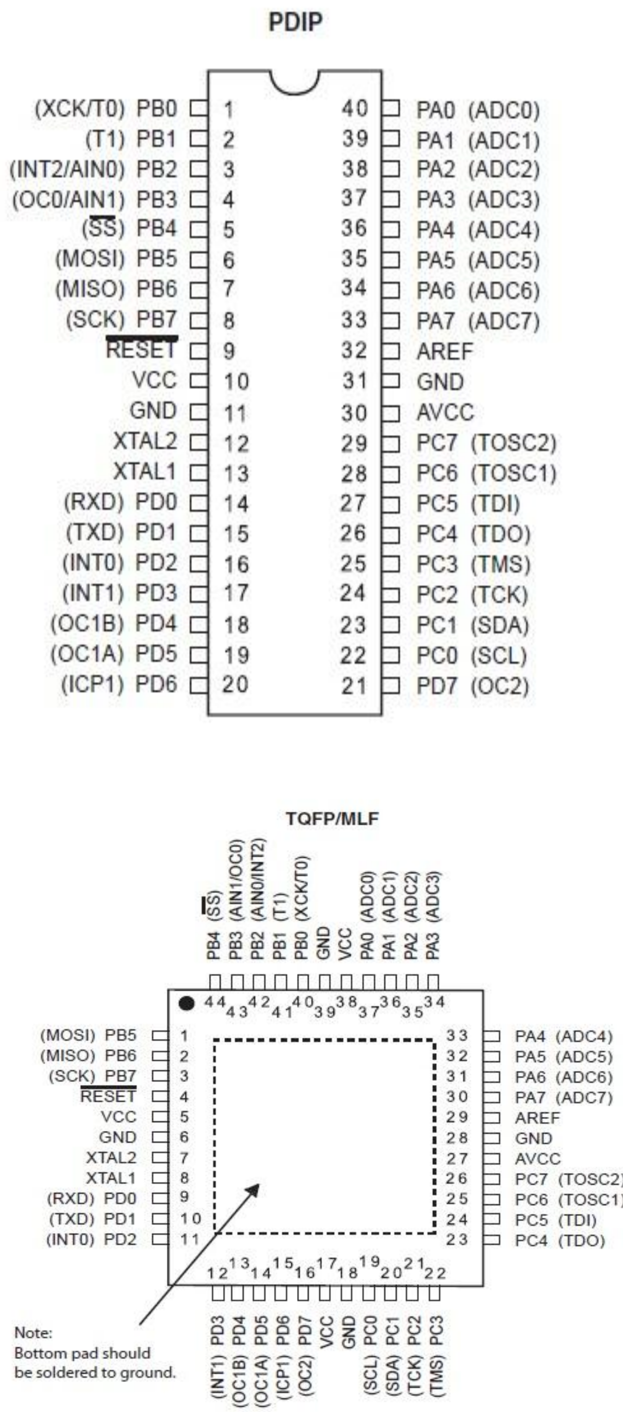

Figure 5.1: pin configuration of ATmega32L[1]

5.3 Need of microcontroller in our system:

As discussed in the project description part, the sole purpose of our project is to determine the time consumed between the two incoming pulses i.e. the time required for a vehicle to cross the distance between the two piezoelectric sensor strips. For determining this time delay we need the help of the timer in microcontroller. Besides that, to detect the incoming pulses from the strips we are going to implement the continuous polling mechanism, for which we need the input/output ports of the microcontroller. So first we need to familiarize ourselves with the functionalities of the input/output ports and the timer and also with their associated registers. So in the coming sections we will concentrate on the working of the timer and I/O ports which will aid us in modeling the microcontroller.

5.4 Input/Output Ports:

5.4.1 Port A (PA0…PA7):

The main job of Port A is to serve as the analog inputs to the A/D Converter. But it can also be used as an 8-bit bi-directional I/O port, if the A/D Converter is not used. Internal pull-up resistors can be provided to each port pin individually[1].

5.4.2 Port B (PB0…PB7):

Originally port B is an 8-bit bi-directional I/O port in which internal pull-up resistors can be provided individually. But Port B also serves many of the other functions of various special features of the ATmega32[1].

5.4.3 Port C (PC0…PC7):

The main function of the port C is to act as an 8-bit bi-directional I/O port with internal pull-up resistors which can be selected for each bit individually. Port C also serves the functions of the JTAG interface and other special features of the ATmega32. If the JTAG interface is enabled, the pull-up resistors on pins PC5(TDI), PC3(TMS) and PC2(TCK) will be activated even if a reset occurs[1].

5.4.4 Port D (PD0…PD7):

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). Port D also serves the functions of various special features of the ATmega32[1].

5.4.5 Registers Associated with the Ports:

Each port pin consists of three register bits: **DDxn**, **PORTxn**, and **PINxn**. The direction of each pin can be configured the DDxn bit in the DDRx Register. If DDxn is written logic one, Pxn is configured as an output pin. If DDxn is written logic zero, Pxn is configured as an input pin.[1]

If PORTxn is written logic one when the pin is configured as an input pin, the pull-up resistor is activated. To switch the pull-up resistor off, PORTxn has to be written logic zero or the pin has to be configured as an output pin[1]. If PORTxn is written logic one when the pin is configured as an output pin, the port pin is driven high (one)[1]. If PORTxn is written logic zero when the pin is configured as an output pin, the port pin is driven low (zero).

5.5 Timer:

In our project work we will be using only Timer0 of the ATmega32L for the simulation purpose. Hence the functionality and associated registers of only Timer0 will be discussed in detail.

5.5.1 8-bit Timer/Counter0:

Timer/Counter0 is a general purpose, single compare unit, 8-bit Timer/Counter module[1]. The main features are:

1. Single Compare Unit Counter
2. Clear Timer on Compare Match (Auto Reload)
3. Glitch-free, Phase Correct Pulse Width Modulator (PWM)
4. Frequency Generator
5. External Event Counter
6. 10-bit Clock Prescaler
7. Overflow and Compare Match Interrupt Sources (TOV0 and OCF0)

5.5.2 Timer/Counter Clock Sources:

An internal or an external clock source can be supplied to the the Timer/Counter by which it can be clocked. The clock select (CS02:0) bits in the Timer/Counter Control Register (TCCR0) are used to select the clock source. The counter is cleared, incremented, or decremented at each timer clock (clkT0) depending on the modes of operation. ClkT0 can be generated from an external or internal clock source, selected by the Clock Select bits (CS02:0). The timer is stopped when no clock source is selected i.e. CS02:0 = 0. However, the

CPU can access TCNT0 value regardless of whether clkT0 is present or not. A CPU write overrides (has priority over) all counter clear or count operations[1].

5.5.3 Output Compare Unit:

The job of the 8-bit comparator in the timer0 is to compare continuously the value in TCNT) register and the value in OCR0 register. Whenever TCNT0 equals OCR0, a match signal is produced by the comparator . In the next timer clock cycle a match will set the Output Compare Flag (OCF0). If enabled (OCIE0 = 1 and Global Interrupt Flag in SREG is set), the Output Compare Flag generates an output compare interrupt[1]. The OCF0 Flag is automatically cleared when the interrupt is executed. Alternatively, the OCF0 Flag can be cleared by software by writing a logical one to its I/O bit location[1].

5.5.4 Registers Associated With Timer/Counter0:

1. Timer/Counter Control Register – TCCR0:

7	6	5	4	3	2	1	0
FOC0	**WGM00**	**COM01**	**COM00**	**WGM01**	**CS02**	**CS01**	**CS00**
W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 5.1: Timer Counter Control Register [1]

Bit 7-FOC0 (Force Output Compare) : The FOC0 bit is only active when the WGM00 bit specifies a non-PWM mode. When writing a logical one to the FOC0 bit, an immediate compare match is forced on the waveform generation unit.

Bit 6, 3 – WGM01:0: Waveform Generation Mode: Counting sequence of the counter is controlled by these bits. Normally Timer/Counter unit supports 4 modes of operation. They are: Normal mode, Clear Timer on Compare Match (CTC) mode and two types of Pulse Width Modulation (PWM) modes.

MODE	WGM01	WGM00	Mode of Operation	TOP	TOV0 Flag set on
1	0	0	Normal	0xFF	MAX
2	0	1	PWM, Phase Correct	0xFF	BOTTOM
3	1	0	CTC	OCR0	MAX
4	1	1	FAST PWM	0xFF	MAX

Table 5.2: functionality of WGM01 and WGM00 bits [1]

Bit 5:4 – COM01:0: Compare Match Output Mode: Output Compare pin (OC0) behavior is defined by these bits. When one or both the bits are set then the normal port functionality of the port pins to which OC0 pin is connected are overridden. These bits are not used in our simulation. So functionality of these bits will not be discussed in detail.

Bit 2:0 – CS02:0: Clock Select: these three bits will be used extensively used in our simulation process. The clock source which will be used by the timer is selected by these three bits.

CS02	CS01	CS00	DESCRIPTION
0	0	0	No clock source(timer stopped)
0	0	1	$Clk_{i/o}$(No Prescaling)
0	1	0	$Clk_{i/o}/8$
0	1	1	$Clk_{i/o}/64$
1	0	0	$Clk_{i/o}/256$
1	0	1	$Clk_{i/o}/1024$
1	1	0	External clock source on T0 pin(on falling edge)
1	1	1	External clock source on T0 pin(on rising edge)

Table 5.3: functionality of CS02, CS01 and CS00 bits [1]

2. Timer/Counter Register – TCNT0:

This 8-bit register holds the value of the Timer/Counter 0. These bits can be read or written by software.

3. Output Compare Register – OCR0:

This 8-bit register contains an 8-bit value with which the timer value is continuously compared. When a compare match occurs, the output compare match flag (OCF0) bit in the Timer/Counter interrupt flag Register is set.

4. Timer/Counter Interrupt Mask Register (TIMSK):

TOIE2	OCIE2	TICIE1	OCIE1A	OCIE1B	TOIE1	OCIE0	TOIE0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 5.4: Timer/Counter Interrupt Mask Register(TIMSK) [1]

Bit 1 – OCIE0: Timer/Counter0 Output Compare Match Interrupt Enable: If the OCIE0 bit is written to 1 as well as the I-bit in the Status Register is set then the Timer/Counter0 Compare Match interrupt is enabled. If a compare match occurs between the timer value and OCR0 register then the corresponding compare match interrupt is executed.

Bit 0 – TOIE0: Timer/Counter0 Overflow Interrupt Enable: When the TOIE0 bit is written to one, and the I-bit in the Status Register is set (one), the Timer/Counter0 Overflow interrupt is enabled. The corresponding interrupt is executed if an overflow in Timer/Counter0 occurs, that is, when the TOV0 bit is set in the Timer/Counter Interrupt Flag Register- TIFR.

5. Timer/Counter Flag Register(TIFR):

OCF2	TOV2	ICF1	OCF1A	OCF1B	TOV1	OCF0	TOV0
R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W

Table 5.5: Timer/Counter Flag Register (TIFR) [1]

Bit 1 – OCF0: Output Compare Flag 0: If a compare match occurs between the value of TCNT0 and the value in OCR0 register then this bit is set. This bit can be cleared by hardware while executing the corresponding interrupt handling vector. It can also be cleared by writing a 1 to this bit.

Bit 0 – TOV0: Timer/Counter0 Overflow Flag 0: If an overflow occurs in Timer/Counter 0 then this bit is set.

CHAPTER 6

SIMULATION OF ATMEGA32L USING AVR STUDIO 4

Simulation of atmega32L using AVR studio 4

AVR Studio 4 is a free Integrated Development Environment (IDE) for writing and debugging AVR applications in Windows 98/XP/ME/2000 and Windows NT environments [2]. AVR Studio 4 provides a project management tool, source file editor, simulator and In-Circuit Emulator interface for the powerful AVR RISC family of microcontrollers. Some of the main features of this tool are Integrated Development Environment, Write, Compile and Debug, Configurable Memory Views (SRAM/EEPROM/Flash/Registers and I/O), Unlimited Number of Break Points, Language support: C, Pascal, BASIC, and Assembly, Extensive Program Flow Control Options etc. [2].

6.1 Concept of the Programming:

In the programming of the microcontroller we need to calculate the milliseconds that are passed between the two incoming pulses. For that we need to select an appropriate clock source and we have to load the Output Compare Register of Timer0 with a proper value so that 1 millisecond will be passed while timer value increases from zero to the value stored in OCR0 register i.e. when compare match occurs 1 millisecond passes. By counting the total no. of milliseconds, we can calculate the total time consumed between two pulses. Distance divided by time will give the required speed of the vehicle.

6.2 Register Values to be Uploaded:

1. DDRB and DDRD registers should be uploaded with the values so that port A and port B will act as input ports for the implementation of polling mechanism. We will be transmitting the alarming signal at 7^{th} bit of port B. So we need to configure the 7^{th} bit of port B as output pin.

Also Port A should be configured as output port so that Port A pins will be used to display the speed. So the value in the DDRB, DDRD and DDRA registers will be

DDRB = 0x80 DDRD = 0x00 DDRA = 0xFF

2. Since the clock frequency of ATmega32L in the simulator (AVR STUDIO 4) is 4MHz we will take the clock input for the timer from the prescaler which divides the input clock cycle by 64. For this purpose we need to configure the CS00, CS01 and CS02 bits in the TCCR0 register.

CS00 = 1 CS01 = 1 CS02 = 0

Also we need to implement the CTC mode of operation for Timer 0. For that we have to configure WGM00 and WGM01 bits in the TCCR0 register.

WGM00 = 0 WGM01 = 1

3. Value to be uploaded in the OCR0 register can be calculated as follows:

Let the length between two strips = 5m.

Frequency of the microcontroller in the simulator = 4 MHz

Frequency after using prescaler = (4/64)MHz = 62500 Hz

62500 pulses ≡ 1 sec.

62.5 pulses ≡ 1 millisec. ≈ 62 pulses

So value in **OCR0** register = 62

6.3 Formula for Speed Calculation:

Let p holds the value of milliseconds that are consumed.

$$\text{Speed} = \frac{5}{p}\ \text{m/millisec.} = \left(\frac{5 \times 1000 \times 3600}{1000 \times p}\right)\ \text{km./hr.} = \frac{5 \times 3600}{p}\ \text{km/hr}$$

6.4 C Programming for the Microcontroller:

```
#include<avr/io.h>
main()
{
DDRB=0x80;                    // Port B set to input port except 7th bit
DDRD=0x00;                    // Port D set to input port
DDRA=0xFF;                    // Port A set to output port
TCCR0=(1<<WGM01);             //CTC mode of operation is selected
OCR0=62;                      // compare register value is loaded
while(1)                      // infinite loop
        {
        int p=0;
        int i=PORTB;          // first pulse is sampled
        while(i==0)           // while no input pulse is detected sample continuously
                {
                i=PORTB;
                }
        TCCR0=(1<<CS01 | 1<<CS00);    // if first input is detected timer is started
        int k= PORTD;                 // second input pulse is sampled
while(k==0)                           //while no second pulse

            {
                int j=TIFR;           // flag register is sampled
```

```
            while(j==0 && k==0)          //while no second input and no flag
                {
                j=TIFR;                   // sample flag register continuously
                k=PORTD;                  // sample second input continuously
                }
            p=p+1;                        // if flag is raised millisecond counter incremented
            TCNT0=0x00;                   // counter value is written to 0
            TIFR=0x02;                    // OCF0 flag is cleared
            k=PORTD;                      // sample second input
            }
        TCCR0=0;                          // if second pulse is detected stop timer
        int x=5*3600/p;                   // speed calculation
        PORTA=x;                          //speed display at Port A
       If(x>50)                           // speed limit = 50
         {
          Setb PORTB7;                    // alarming signal is set on bit 7 of Port B
         }
        }
}
```

6.5 Screenshots of the Simulation:

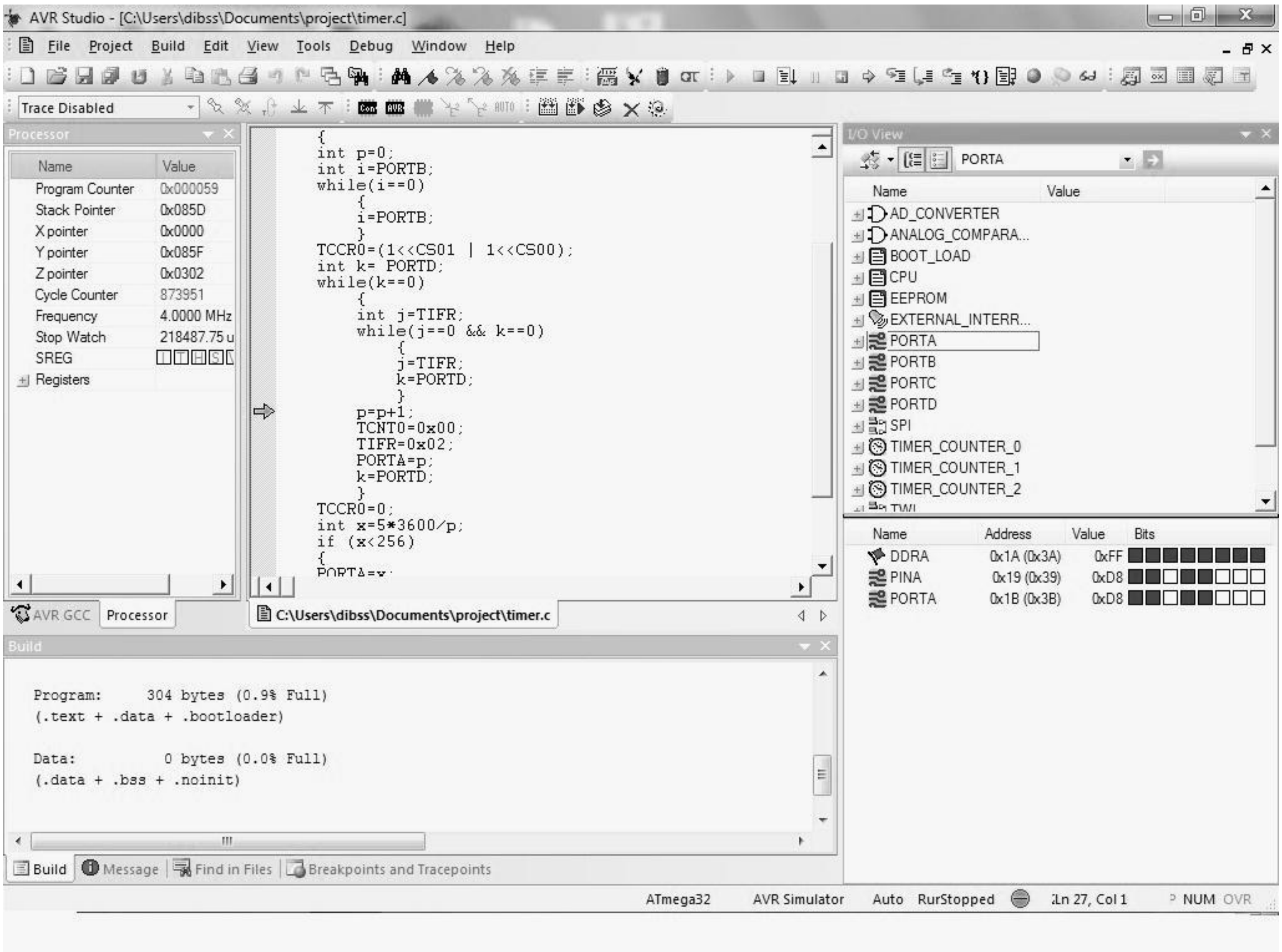

Figure 6.1: Screen shot 1

The above simulation is run for 216 milliseconds. In the above programming Port A is configured to show the value of the variable "p" as long as speed is not calculated. As we know the variable "p" holds the value of the milliseconds passed. We can see the value shown by the Port A register at the lower part of the right hand side of the screen. The value shown is "D8" which is equal to 216 milliseconds. In the left hand side of the screen, in the processor window we can see the reading of the stop watch to be around 218 milliseconds which is nearly equal to the value of the variable p.

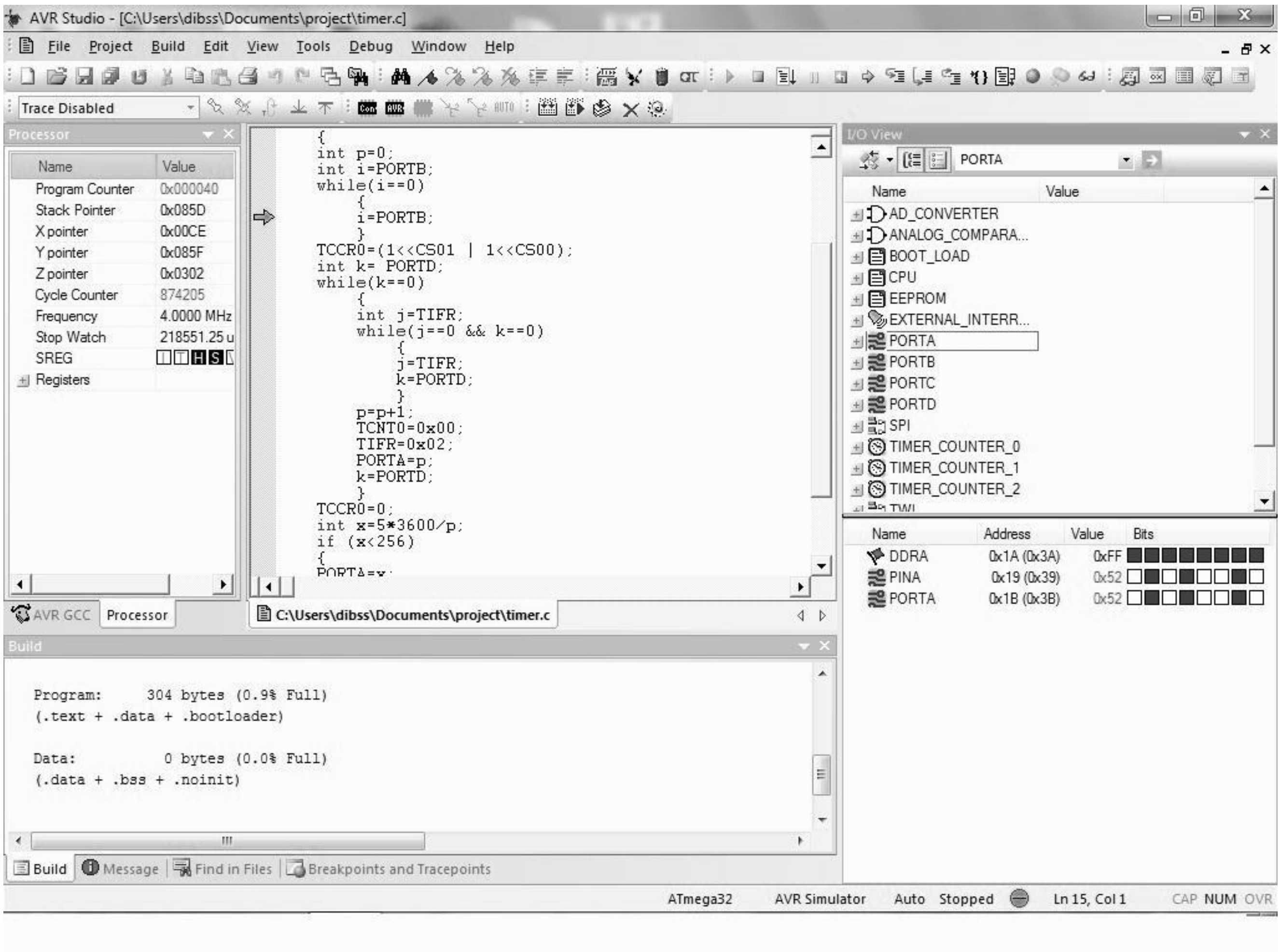

Figure 6.2: Screen shot 2

In the above screen shot we can see the value of the speed at Port A. The above simulation is run for 216 milliseconds. So by the formula derived for the speed of the vehicle the speed is found to be 80km/hr. The Port A shows the hexadecimal value of this speed as 52.

6.6 Observations:

Simulation no.	Value of "p" in HEX	Value of "p" in DEC (milliseconds passed)	Speed value observed (in km/hr)
1	D8	216	80
2	9B	155	116
3	FF	255	70
4	BF	191	94
5	E6	230	78

Table 6.1: Observation Table

The simulations are run for 5 times and above results were noted. It was observed that the values of the stop watch were nearly equal to that of the variable "p" which holds the milliseconds value.

Chapter 7

Conclusion

Conclusion

With the progress and development of society it is extremely important for mankind to ensure the security of human life which is very much precious. An attempt is made by us to harness the property of piezoelectric material to develop a model that can be useful as a security system in the society. May the dynamic characteristic of piezoelectric material can make some problem while two vehicle crosses simultaneously or continuously with little time difference or the robustness of the model may create some problem but while industrial production these lacunas can be taken care of easily. Also the cost of individual system can be minimized when these can be produced in a mass industrially.

7.1 Scope for the future:

The number of vehicle will definitely increase in future. May the resources like petrol and diesel get vanish but there will definitely be an alternative for transportation like the vehicle may run through electric charge or sea water or any other alternative but people will always use cars, bikes, buses for their transportation. In present time the MIGs (middle income group) can also afford money for cars. The Nano cars produced by Tata group got a very low cost also. So in future the vehicle cost will definitely reduce a lot due to heavy competition between companies in this domain.

So there always be a chance of accident and we can't neglect it. The total area is fixed but the population is increasing exponentially. So in future there will be more roads, more vehicles and subsequently more accidents and for such a big number of roads we can't afford that much number of radar or laser guns. Present time the accidents death pole in India is

1,30,000 in a year. This seems really unfair as this is human life and a single death takes another 4 souls of his family with him. So we can't afford to lose such a number of lives.

To stop accidents accident what we can do is that we may reduce the number of vehicles but it seems impossible on its part. So what we can do is that we can impose certain rules for driving purpose. Accidents can be control by controlling the speed of vehicles. A vehicle should not be allowed to move with more than a specific speed on a certain roads. Example – a car can't move with a velocity more than 30 km/hr. on a road near to schools as it may harm children life. Or it can't move with the same velocity near a hospital etc.

The above proposed model can solve this problem in future quite efficiently. We can use this concept of piezoelectric sensor to build the proposed sensor and can use it on road for safety purpose. On Preliminary basis we have designed a small model to counter the problem of accidents. This model contains only two piezoelectric sensor only to implement the concept. But this model can be a great asset in future due to its low cost. If we manufacture it on a mass then its cost will further reduce a lot.

In this model we have used only a buzzer to activate the alarming condition to alert the human operator or the traffic police on the next square but it seems unsatisfactory to recognize the identity of vehicle crossed the strips with higher velocity. Only from the buzzer sound the vehicle may not be recognized as if two vehicle passes on the arrays of sensor one after another continuously. The buzzer sound may be inefficient to tell which vehicle crossed with higher velocity. So in future the model can be modified to recognize exact vehicle which disobeyed the rule. It can be done by attaching a digital camera focusing just above the second sensor array. The camera will be trigger on along with the buzzer. So the photo of the number plate of the vehicle can be taken to recognize the vehicle. This will be a great advantage as it will give a direct number plate photo to the security substation.

In future this model can be implemented section wise. So that if two vehicles passes simultaneously on the arrays then two separate sections can identify which section vehicle got higher velocity and which has normal or else both crossed with higher velocity. Two separate sections may connect to different microcontroller and can perform independently.

Now the question comes is that can this sensors sustain such a higher weight of vehicles passes over it. While producing in a mass special concentration can be given to their robustness. It can be protected from higher weight by using iron like metals as a shield or other advanced metal can be used to protect these sensors from external environmental effect. But again special care is needed so that the sensor doesn't lose its sensitivity.

7.2 Summary:

Instead of taking arrays of piezoelectric sensors we have taken two sensors for designing of our model. As its clear from above project work that the concept holds good in case of 2 piezoelectric sensor, hence its surely possible to design the practical and working version of this model. The above project is only up to the simulation part but the real time implementation will be a little tougher as it includes all real time environmental problems and artifacts.

References

[1] ATmega32/L Datasheet

Available: http://www.atmel.com/atmel/acrobat/doc2503.pdf

[2] http://www.efo.ru/ftp/pub/atmel/_AVR_MCUs_8bit/_Technical_Library/software/studio4/index.html

[3] http://www.aurelienr.com/electronique/piezo/piezo.pdf

[4] http://www.pc-control.co.uk/piezoelectric_effect.html

[5] http://www.microdigitaled.com/AVR/Software/AVRstudioTutorial.pdf

[6] http://pdf1.alldatasheet.com/datasheet-pdf/view/22615/STMICROELECTRONICS/L7805.html

[7] www.aarewah.com/uploads/TechnoCraft/**AstronomyClub**Report.doc

[8] Bentley john p. principle of measurement systems. Pearson education. 4th edition 2009 .