Digital Thermometer using ATmega8 Microcontroller

Technical Report · September 2013

DOI: 10.13140/2.1.3551.7763

Utsho A Arefín
Northern University Bangladesh

Digital Thermometer using ATmega8 Microcontroller

Submitted by

Vaskar Roy
ID: ECE-070300090

Md. Riazul Islam Sagar
ID: ECE-070300086

Supervised by

Ashraful Arefin
Sr. Lecturer
Electrical & Electronic Engineering

SEPTEMBER 2013

Department of Electronics & Communication Engineering
NORTHERN UNIVERSITY BANGLADESH

APPROVAL

The project report titled “Digital Thermometer using ATmega8 Microcontroller”, submitted by Vaskar Roy (ID: ECE-070300090) & Md. Riazul Islam Sagar (ID: ECE-070300086) students of Electronics & Communication Engineering, Northern University Bangladesh, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science (Engineering) in Electronics & Communication Engineering of the year 2013 and approved to its style and contents.

Board of Examination

1) ……………………………………

Ashraful Arefin (Supervisor)
Sr. Lecturer
Department of Electrical & Electronic Engineering
Northern University Bangladesh

2) ………………………………………

Lecturer
Department of Electrical & Electronic Engineering
Northern University Bangladesh

3) ………………………………………

Lecturer
Department of Electrical & Electronic Engineering
Northern University Bangladesh

4) ………………………………………

Professor Dr. Md. Shah Alam
Dean
Faculty of Science & Engineering
Northern University Bangladesh

DECLARATION

We, hereby, declare that the work presented in this project is the outcome of the project work performed by us under the supervision of Ashraful Arefin Sr. Lecturer, Department of Electrical & Electronic Engineering, Northern University Bangladesh. We also declare that no part of this project has been submitted elsewhere for the award of any degree.

Countersigned **Signature**

…………………………..

Supervisor,
Ashraful Arefin
Sr. Lecturer
Department of Electrical & Electronics Engineering
Northern University Bangladesh

…………………………

Vaskar Roy
ID: ECE-070300090
Dept. of ECE

……………………………

Md. Riazul Islam Sagar
ID: ECE-070300086
Dept. of ECE

ACKNOWLEDGEMENT

In the world no work has been completed smoothly where there is no guidance and help. The project work is the result of an effort of more than four month duration during which we have been accompanied and supported by many people. It is a pleasure that we have now the opportunity to express our gratitude to all of them.

Firstly we offer millions thanks to the Almighty, who has given us strength to complete of this project successfully. At the very beginning, we would like to express our immense gratitude and delightful thanks to our supervisor Ashraful Arefin, Sr. Lecturer, Department of Electrical & Electronic Engineering, Northern University Bangladesh for giving us an opportunity to work on this topic, in which we found interest compared to conventional course works. Our respect is also to him for his wisdom, guidance, supervision and faithful discussion with us throughout the work. He, not only agrees to supervise the project wholeheartedly, but also from the beginning of our study here. He supported us with incessant generosity. We would also like to extend our warmest thanks to as well as the officers and stuffs related to our departmental laboratories.

Finally big thanks to our parents, to our family and friends for their invaluable encouragement and support all the way.

Vaskar Roy
&
Md. Riazul Islam Sagar
September, 2013

ABSTRACT

Global warming is a big problem now a day. People want to be more concerned about the weather. They want to know the temperature and humidity thus they can forecast the weather change. Here digital thermometer based on ATmega8 microcontroller is a big deal. That can sense the temperature from the nature and provide digital output. LCD display can show that result, computer can analyze it and forecast that day's weather. This result can be applied to airplane, ship, bus, industry etc.

This project based on a microcontroller and a temperature sensor has an analog sensor that converts the surrounding's temperature to a proportional analog voltage. The output from the sensor is connected to one of the ADC channel inputs of the microcontroller to derive the equivalent temperature value in digital format. The computed temperature is displayed in a character LCD in degree centigrade.

As it is not so costly, industrially that can be spread out vastly. The day is not so far, when people use that types of device for their daily purpose. And that was our object to construct such a device which can be operated easily and get output smoothly as a normal people can use it.

So to construct a user friendly device which can sense the temperature from the nature and cost a little is our object.

Contents

Chapter 1 Page No
Introduction

Chapter 2
Temperature Sensor: LM35

Chapter 3
ATmega8 Microcontroller

Chapter 1
Introduction

1.1 Thermometer

A thermometer comes from the Greek word, *thermos*, meaning "hot" and *metron*, "measure". It is a device that measures temperature or temperature gradient using a variety of different principles. A thermometer has two important elements: the temperature sensor (e.g. the bulb on a mercury-in-glass thermometer) in which some physical change occurs with temperature, plus some means of converting this physical change into a numerical value (e.g. the visible scale that is marked on a mercury-in-glass thermometer).

Thermometers measure temperature, by using materials that change in some way when they are heated or cooled. In a mercury or alcohol thermometer the liquid expands as it is heated and contracts when it is cooled, so the length of the liquid column is longer or shorter depending on the temperature. Modern thermometers are calibrated in standard temperature units such as Fahrenheit (used in the United States) or Celsius (used in Canada) and Kelvin (used mostly by scientists).

Figure 1.1: Thermometer

1.2 Temperature

While an individual thermometer is able to measure degrees of hotness, the readings on two thermometers cannot be compared unless they conform to an agreed scale. There is today an absolute thermodynamic temperature scale. Internationally agreed temperature scales are designed to approximate this closely, based on fixed points and interpolating thermometers. The most recent official temperature scale is the International Temperature Scale of 1990. It extends from 0.65 K (−272.5 °C; −458.5 °F) to approximately 1,358 K (1,085 °C; 1,985 °F).

1.3 Development

Various authors have credited the invention of the thermometer to Cornelis Drebbel, Robert Fludd, Galileo Galilei or Santorio Santorio. The thermometer was not a single invention, however, but a development.Philo of Byzantium and Hero of Alexandria knew of the principle that certain substances, notably air, expand and contract and described a demonstration in which a closed tube partially filled with air had its end in a container of water. The expansion and contraction of the air caused the position of the water/air interface to move along the tube. Such a mechanism was later used to show the hotness and coldness of the air with a tube in which the water level is controlled by the expansion and contraction of the gas. These devices were developed by several European scientists in the 16th and 17th centuries, notably Galileo Galilei. As a result, devices were shown to produce this effect reliably, and the term *thermoscope* was adopted because it reflected the changes in sensible heat (the concept of temperature was yet to arise). The difference between a thermoscope and a thermometer is that the latter has a scale. Though Galileo is often said to be the inventor of the thermometer, what he produced were thermoscopes. The first clear diagram of a thermoscope was published in 1617 by Giuseppe Biancani: the first showing a scale and thus constituting a thermometer was by Robert Fludd in 1638. This was a vertical tube, closed by a bulb of air at the top, with the lower end opening into a vessel of water. The water level in the tube is controlled by the expansion and contraction of the air, so it is what we would now call an air thermometer. The first person to put a scale on a thermoscope is variously said to be Francesco Sagredo or Santorio Santorioin about 1611 to 1613.

Figure 1.2: Galileo Thermometer 24 degrees

The word thermometer (in its French form) first appeared in 1624 in *La Récréation Mathématique* by J. Leurechon, who describes one with a scale of 8 degrees. The above instruments suffered from the disadvantage that they were also barometers, i.e. sensitive to air pressure. In about 1654 Ferdinando II de' Medici, Grand Duke of Tuscany, made sealed tubes part-filled with alcohol, with a bulb and stem; the first modern-style thermometer, dependent on the expansion of a liquid, and independent of air pressure. Many other scientists experimented with various liquids and designs of thermometer. However, each inventor and each thermometer was unique—there was no standard scale. In 1665 Christiaan Huygens suggested using the melting and boiling points of water as standards, and in 1694 Carlo Renaldini proposed using them as fixed points on a universal scale. In 1701 Isaac Newton proposed a scale of 12 degrees between the melting point of ice and body temperature. Finally in 1724 Daniel Gabriel Fahrenheit produced a temperature scale which now (slightly adjusted) bears his name. He could do this because he manufactured thermometers, using mercury (which has a high coefficient of expansion) for the first time and the quality of his production could provide a finer scale and greater reproducibility, leading to its general adoption. In 1742 Anders Celsius proposed a scale with zero at the boiling point and 100 degrees at the freezing point of water, though the scale which now bears his name has them the other way around.[10]In 1866 Sir Thomas Clifford Allbutt invented a clinical thermometer that produced a body temperature reading in five minutes as opposed to twenty.[11] In 1999 Dr. Francesco Pompei of the Exergen Corporation introduced the world's first temporal artery thermometer, a non-invasive temperature sensor which scans the forehead in about two seconds and provides a medically accurate body temperature.

Old thermometers were all non-registering thermometers. That is, the thermometer did not hold the temperature after it was moved to a place with a different temperature. Determining the temperature of a pot of hot liquid required the user to leave the thermometer in the hot liquid until after reading it. If the non-registering thermometer was removed from the hot liquid, then the temperature indicated on the thermometer would immediately begin changing to reflect the temperature of its new conditions (in this case, the air temperature). Registering thermometers are designed to hold the temperature indefinitely, so that the thermometer can be removed and read at a later time or in a more convenient place. The first registering thermometer was designed and built by James Six in 1782, and the design, known as Six's thermometer is still in wide use today. Mechanical registering thermometers hold either the highest or lowest temperature recorded, until manually re-set, e.g., by shaking down a mercury-in-glass thermometer, or until an even more extreme temperature is experienced. Electronic registering thermometers may be designed to remember the highest or lowest temperature, or to remember whatever temperature was present at a specified point in time. Thermometers increasingly use electronic means to provide a digital display or input to a computer.

1.4 Physical Principles of Thermometry

Thermometers may be described as empirical or absolute. Absolute thermometers are calibrated numerically by the thermodynamic absolute temperature scale. Empirical thermometers are not in general necessarily in exact agreement with absolute thermometers as to their numerical scale readings, but to qualify as thermometers at all they must agree with absolute thermometers and with each other in the following way: given any two bodies isolated in their separate respective thermodynamic equilibrium states, all thermometers agree as to which of the two has the higher temperature, or that the two have equal temperatures.[14] For any two empirical thermometers, this does not require that the relation between their numerical scale readings be linear, but it does require that relation to be strictly monotonic.[15] This is a fundamental character of temperature and thermometers. As it is customarily stated in textbooks, taken alone, the so-called "zeroth law of thermodynamics" fails to deliver this information, but the statement of the zeroth law of thermodynamics by James Serrin in 1977, though rather mathematically abstract, is more informative for thermometry: "Zeroth Law – There exists a topological line M which serves as a coordinate manifold of material behaviour. The points L of the manifold M are called 'hotness levels', and M is called the 'universal hotness manifold'. To this information there needs to be added a sense of greater hotness; this sense can be had, independently of calorimetry, of thermodynamics, and of properties of particular materials, from Wien's displacement law of thermal radiation: the temperature of a bath of thermal radiation is proportional, by a universal constant, to the frequency of the maximum of its frequency spectrum; this frequency is always positive, but can have values that tend to zero. Another way of identifying hotter as opposed to colder conditions is supplied by Planck's principle, that when a process of isochoric adiabatic work is the sole means of change of internal energy of a closed system, the final state of the system is never colder than the initial state; except for phase changes with latent heat, it is hotter than the initial state. There are several principles on which empirical thermometers are built, as listed in the section of this article entitled "Primary and secondary thermometers". Several such principles are essentially based on the constitutive relation between the state of a suitably selected particular material and its temperature. Only some materials are suitable for this purpose, and they may be considered as "thermometric materials". Radiometric thermometry, in contrast, can be only very slightly dependent on the constitutive relations of materials. In a sense then, radiometric thermometry might be thought of as "universal". This is because it rests mainly on a universality character of thermodynamic equilibrium, that it has the universal property of producing blackbody radiation.

1.5 Primary and secondary thermometers

Thermometers can be divided into two separate groups according to the level of knowledge about the physical basis of the underlying thermodynamic laws and quantities. For **primary thermometers** the measured property of matter is known so well that temperature can be calculated without any unknown quantities. Examples of these are thermometers based on the equation of state of a gas, on the velocity of sound in a gas, on the thermal noise (see Johnson–Nyquist noise), voltage or current of an electrical resistor, on blackbody radiation, and on the angular anisotropy of gamma ray emission of certain radioactive nuclei in a magnetic field. Primary thermometers are relatively complex.

Secondary thermometers are most widely used because of their convenience. Also, they are often much more sensitive than primary ones. For secondary thermometers knowledge of the measured property is not sufficient to allow direct calculation of temperature. They have to be calibrated against a primary thermometer at least at one temperature or at a number of fixed temperatures. Such fixed points, for example, triple points and superconducting transitions, occur reproducibly at the same temperature.

1.6 Calibration

Thermometers can be calibrated either by comparing them with other calibrated thermometers or by checking them against known fixed points on the temperature scale. The best known of these fixed points are the melting and boiling points of pure water. (Note that the boiling point of water varies with pressure, so this must be controlled.)

The traditional method of putting a scale on a liquid-in-glass or liquid-in-metal thermometer was in three stages:

1. Immerse the sensing portion in a stirred mixture of pure ice and water at 1 Standard atmosphere (101.325 kPa; 760.0 mmHg) and mark the point indicated when it had come to thermal equilibrium.
2. Immerse the sensing portion in a steam bath at 1 Standard atmosphere (101.325 kPa; 760.0 mmHg) and again mark the point indicated.
3. Divide the distance between these marks into equal portions according to the temperature scale being used.

Other fixed points used in the past are the body temperature (of a healthy adult male) which was originally used by Fahrenheit as his upper fixed point (96 °F (36 °C) to be a number divisible by 12) and the lowest temperature given by a mixture of salt and ice, which was originally the definition of 0 °F (−18 °C).[32] (This is an example of a Frigorific mixture). As body temperature varies, the Fahrenheit scale was later changed to use an upper fixed point of boiling water at 212 °F (100 °C).These have now been replaced by the defining points in the International Temperature Scale of 1990, though in practice the melting point of water is more commonly used than its triple point, the latter being more difficult to manage and thus restricted to critical standard measurement. Nowadays manufacturers will often use a thermostat bath or solid block where the temperature is held constant relative to a calibrated thermometer. Other thermometers to be calibrated are put into the same bath or block and allowed to come to equilibrium, then the scale marked, or any deviation from the instrument scale recorded.[34] For many modern devices calibration will be stating some value to be used in processing an electronic signal to convert it to a temperature.

1.7 Uses

Thermometers utilize a range of physical effects to measure temperature. Temperature sensors are used in a wide variety of scientific and engineering applications, especially measurement systems. Temperature systems are primarily either electrical or mechanical, occasionally inseparable from the system which they control (as in the case of a mercury-in-glass thermometer). Thermometers are used in roadways in cold weather climates to help determine if icing conditions exist. Indoors, thermistors are used in climate control systems such as air conditioners, freezers, heaters, refrigerators, and water heaters.[39] Galileo thermometers are used to measure indoor air temperature, due to their limited measurement range.

Alcohol thermometers, infrared thermometers, mercury-in-glass thermometers, recording thermometers, thermistors, and Six's thermometers are used in meteorology and climatology in various levels of the atmosphere and oceans. Aircraft use thermometers and hygrometers to determine if atmospheric icing conditions exist along their flight path. These measurements are used to initialize weather forecast models. Thermometers are used in roadways in cold weather climates to help determine if icing conditions exist and indoors in climate control systems.

Bi-metallic stemmed thermometers, thermocouples, infrared thermometers, and thermistors are handy during cooking in order to know if meat has been properly cooked. Temperature of food is important because if it sits in environments with a temperature between 5 and 57 °C (41 and 135 °F) for four hours or more, bacteria can multiply leading to foodborne illnesses. Thermometers are used in the production of candy.

Medical thermometers such as mercury-in-glass thermometers,[40] infrared thermometers,[41] pill thermometers, and liquid crystal thermometers are used in health care settings to determine if individuals have a fever or are hypothermic.

Such liquid crystal thermometers (which use thermochromic liquid crystals) are also used in mood rings and used to measure the temperature of water in fish tanks.

Fiber Bragg grating temperature sensors are used in nuclear power facilities to monitor reactor core temperatures and avoid the possibility of nuclear meltdowns.

A thermometer constructed for probing stored food is also called a "temperature wand".

1.8 Various types of thermometer

- Alcohol thermometer
- Mercury-in-glass thermometer
- Balco alloy
- Beckmann differential thermometer
- Bi-metal mechanical thermometer
- Coulomb blockade thermometer
- Galileo thermometer
- Heat meter
- Infrared thermometer
- Digital thermometer

1.9 Digital Thermometer

A digital thermometer is an instrument that measures the body temperature and it uses electronic means to provide a digital display of the temperature reading. Such thermometers can also provide a digital input to a computer.

Figure 1.3: Digital Thermometer

There are different ways to construct digital Thermometer:

- Using Amplifier and V/F converter.
- Using CA-3161 IC and transistor.
- Using 1N4148 diode and ICs.
- Using microcontroller

Microcontroller is a good choice because of it is the latest and reliable device about this purpose. Also, it provides an opportunity to learn using sensors to measure the real world signals that are analog in nature.

To complete the construction we have used resistor, capacitor, Temperature sensor device, LCD display and transformer along with the microcontroller.

Chapter 2
Temperature Sensor: LM35

2.1 Description

The LM35 is an integrated circuit sensor that can be used to measure temperature. It is a precision IC temperature sensor with its output proportional to the temperature (in ^{o}C). The sensor circuitry is sealed and therefore it is not subjected to oxidation and other processes. With LM35, temperature can be measured more accurately than with a thermistor. It also possess low self-heating and does not cause more than 0.1^{o}C temperature rise in still air. The operating temperature range is from -55°C to 150°C. The output voltage varies by 10mV in response to every ^{o}C rise/fall in ambient temperature, *i.e.*, its scale factor is 0.01V/ ^{o}C.

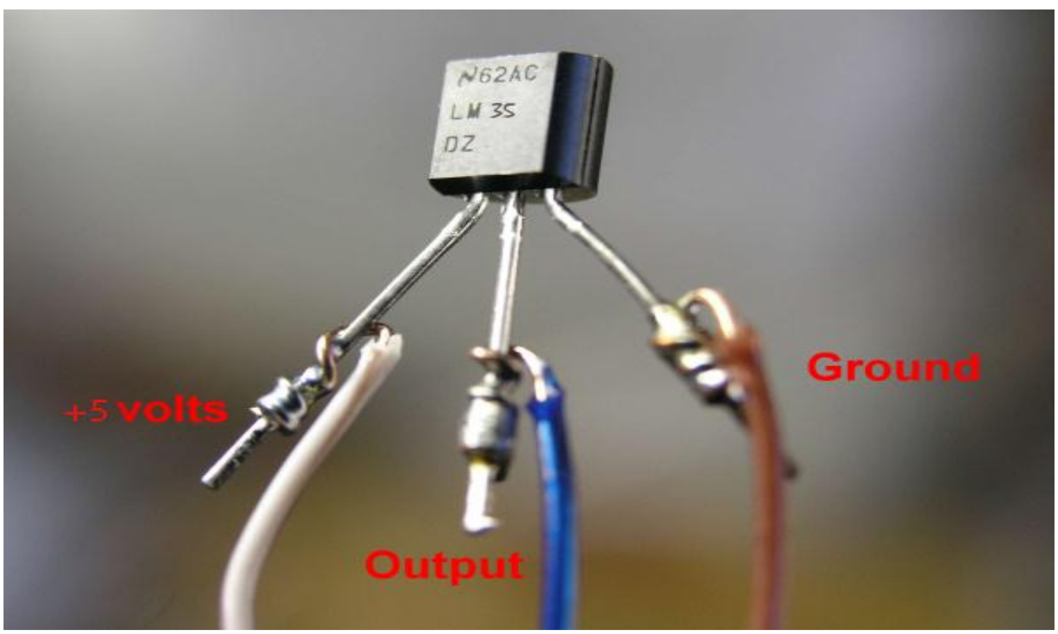

Figure 2.1: LM-35 Sensor

2.2 CONNECTION DIAGRAMS

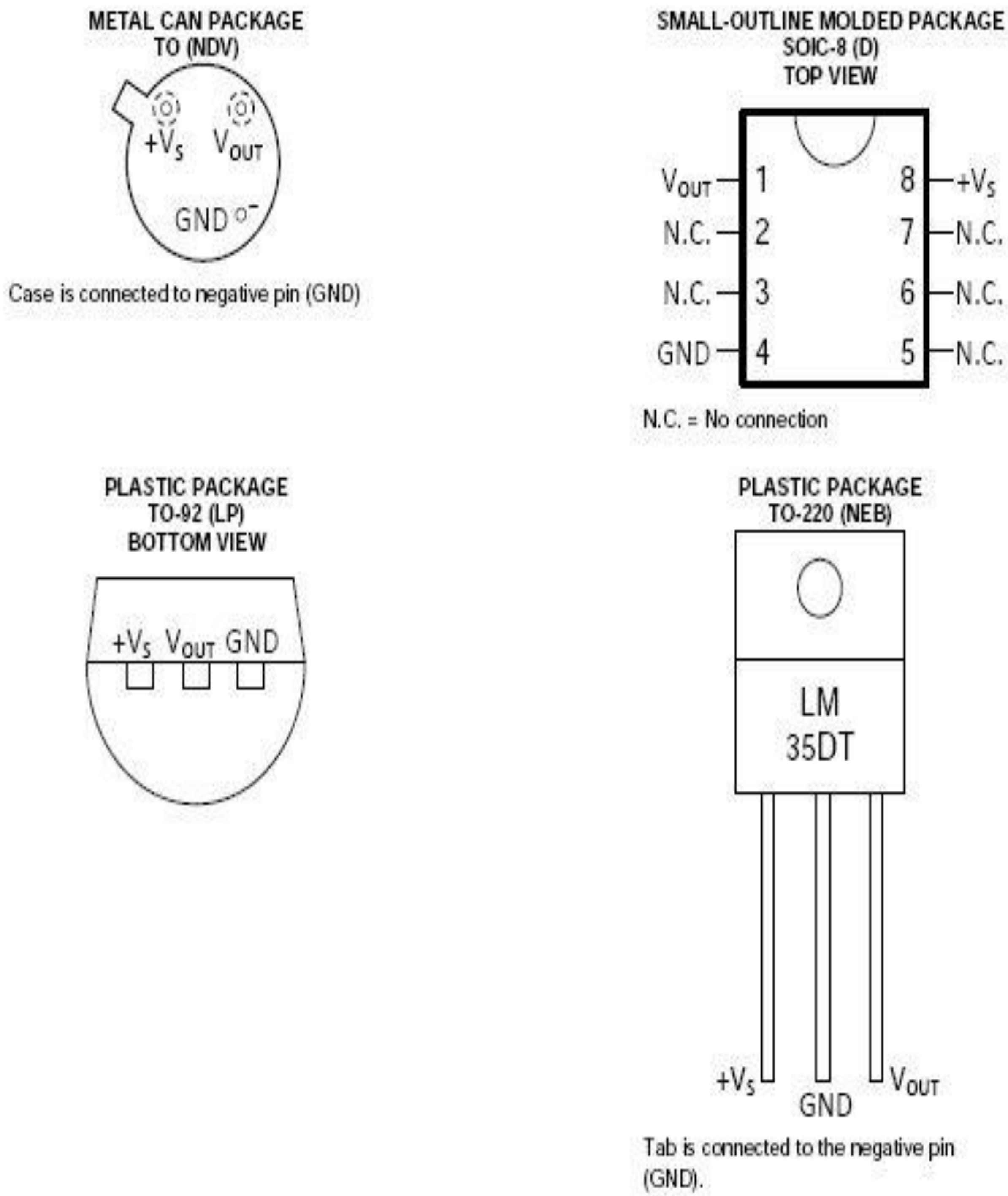

Figure 2.2: Connection Diagram.

2.3 Pin Description of LM35:

Pin No	Function	Name
1	Supply voltage: +5V	Vcc
2	Output voltage (+6V to -1V)	Output
3	Ground (0V)	Ground

Table 2.2: Pin Description of LM35

2.4 Features

- Calibrated directly in ^{o}C .
- Linear +10mV/ ^{o}C scale factor.
- 0.5^{o}C ensured accuracy (at +25^{o}C).
- Rated for full -55^{o}C to + 150^{o}C .
- Suitable for remote applications.
- Low cost due to wafer-level trimming.
- Operates from 4 to 30 V.
- Less than 60 μF current drain.
- Low self-heating, 0.08^{o}C in still air.

Chapter 3
ATmega8 Microcontroller

3.1 Microcontroller

A microcontroller is a small computer on a single integrated circuit containing a processor core, memory, and programmable input/output peripherals. Program memory in the form of NOR flash or OTP ROM is also often included on chip, as well as a typically small amount of RAM. Microcontrollers are designed for embedded applications, in contrast to the microprocessors used in personal computers or other general purpose applications.

Microcontrollers are used in automatically controlled products and devices, such as automobile engine control systems, implantable medical devices, remote controls, office machines, appliances, power tools, toys and other embedded systems. By reducing the size and cost compared to a design that uses a separate microprocessor, memory, and input/output devices, microcontrollers make it economical to digitally control even more devices and processes. Mixed signal microcontrollers are common, integrating analog components needed to control non-digital electronic systems.

Some microcontrollers may use four-bit words and operate at clock rate frequencies as low as 4 kHz, for low power consumption (single-digit mille-watts or microwatts). They will generally have the ability to retain functionality while waiting for an event such as a button press or other interrupt; power consumption while sleeping (CPU clock and most peripherals off) may be just Nano-watts, making many of them well suited for long lasting battery applications. Other microcontrollers may serve performance-critical roles, where they may need to act more like a digital signal processor (DSP), with higher clock speeds and power consumption.

3.2 History

The first single-chip microprocessor was the 4-bit Intel 4004 released in 1971, with the Intel 8008 and other more capable microprocessors becoming available over the next several years. However, both processors required external chips to implement a working system, raising total system cost, and making it impossible to economically computerize appliances.

The Smithsonian Institution says TI engineers Gary Boone and Michael Cochran succeeded in creating the first microcontroller in 1971. The result of their work was the TMS 1000, which went commercial in 1974. It combined read-only memory, read/write memory, processor and clock on one chip and was targeted at embedded systems.

Partly in response to the existence of the single-chip TMS 1000,[2] Intel developed a computer system on a chip optimized for control applications, the Intel 8048, with commercial parts first shipping in 1977.[2] It combined RAM and ROM on the same chip. This chip would find its way into over one billion PC keyboards, and other numerous applications. At that time Intel's President, Luke J. Valenter, stated that the microcontroller was one of the most successful in the company's history, and expanded the division's budget over 25%.

Most microcontrollers at this time had two variants. One had an erasable EPROM program memory, with a transparent quartz window in the lid of the package to allow it to be erased by exposure to ultraviolet light. The other was a PROM variant which was only programmable once; sometimes this was signified with the designation OTP, standing for "one-time programmable". The PROM was actually exactly the same type of memory as the EPROM, but because there was no way to expose it to ultraviolet light, it could not be erased. The erasable versions required ceramic packages with quartz windows, making them significantly more expensive than the OTP versions, which could be made in lower-cost opaque plastic packages. For the erasable variants, quartz was required, instead of less expensive glass, for its transparency to ultraviolet—glass is largely opaque to UV—but the main cost differentiator was the ceramic package itself.

In 1993, the introduction of EEPROM memory allowed microcontrollers (beginning with the Microchip PIC16x84 to be electrically erased quickly without an expensive package as required for EPROM, allowing both rapid prototyping, and In System Programming. (EEPROM technology had been available prior to this time, but the earlier EEPROM was more expensive and less durable, making it unsuitable for low-cost mass-produced microcontrollers.) The same year, Atmel introduced the first microcontroller using Flash memory, a special type of EEPROM. Other companies rapidly followed suit, with both memory types.

Cost has plummeted over time, with the cheapest 8-bit microcontrollers being available for under 0.25 USD in quantity (thousands) in 2009, and some 32-bit microcontrollers around 1 USD for similar quantities.

Nowadays microcontrollers are cheap and readily available for hobbyists, with large online communities around certain processors.

In the future, MRAM could potentially be used in microcontrollers as it has infinite endurance and its incremental semiconductor wafer process cost is relatively low.

3.3 ATmega8 Microcontroller

The Atmel® AVR® ATmega8 is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega8 achieves through puts approaching 1MIPS per MHz, allowing the system designed to optimize power consumption versus processing speed.

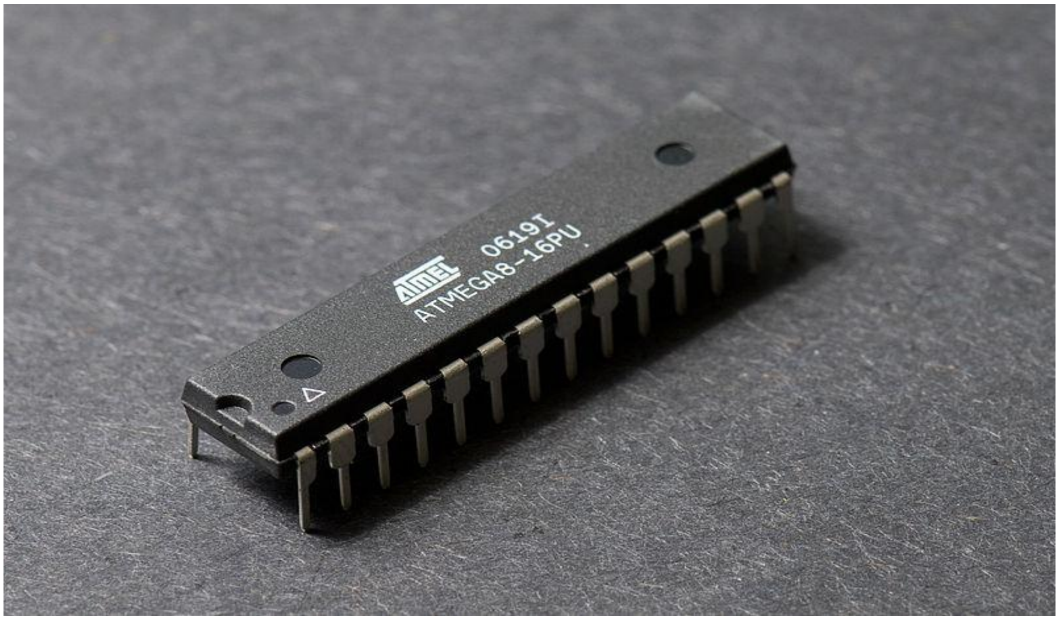

Figure 3.1: ATmega8 Microcontroller

3.4 Features

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

• Advanced RISC Architecture

– 130 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

– 8Kbytes of In-System Self-programmable Flash program memory

– 512Bytes EEPROM

– 1Kbyte 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

• Peripheral Features

– Real Time Counter with Separate Oscillator

– Three PWM Channels

– 8-channel ADC in TQFP and QFN/MLF package

Eight Channels 10-bit Accuracy

– 6-channel ADC in PDIP package

Six Channels 10-bit Accuracy

– 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

– External and Internal Interrupt Sources

– Five Sleep Modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby

• I/O and Packages

– 23 Programmable I/O Lines

– 28-lead PDIP, 32-lead TQFP, and 32-pad QFN/MLF

• Operating Voltages

– 4.5V - 5.5V (ATmega8)

• Speed Grades

– 0 - 8MHz (ATmega8L)

• Power Consumption at 4Mhz, 3V, 25□C

– Active: 3.6mA

– Idle Mode: 1.0mA

– Power-down Mode: 0.5

3.5 Pin Configurations

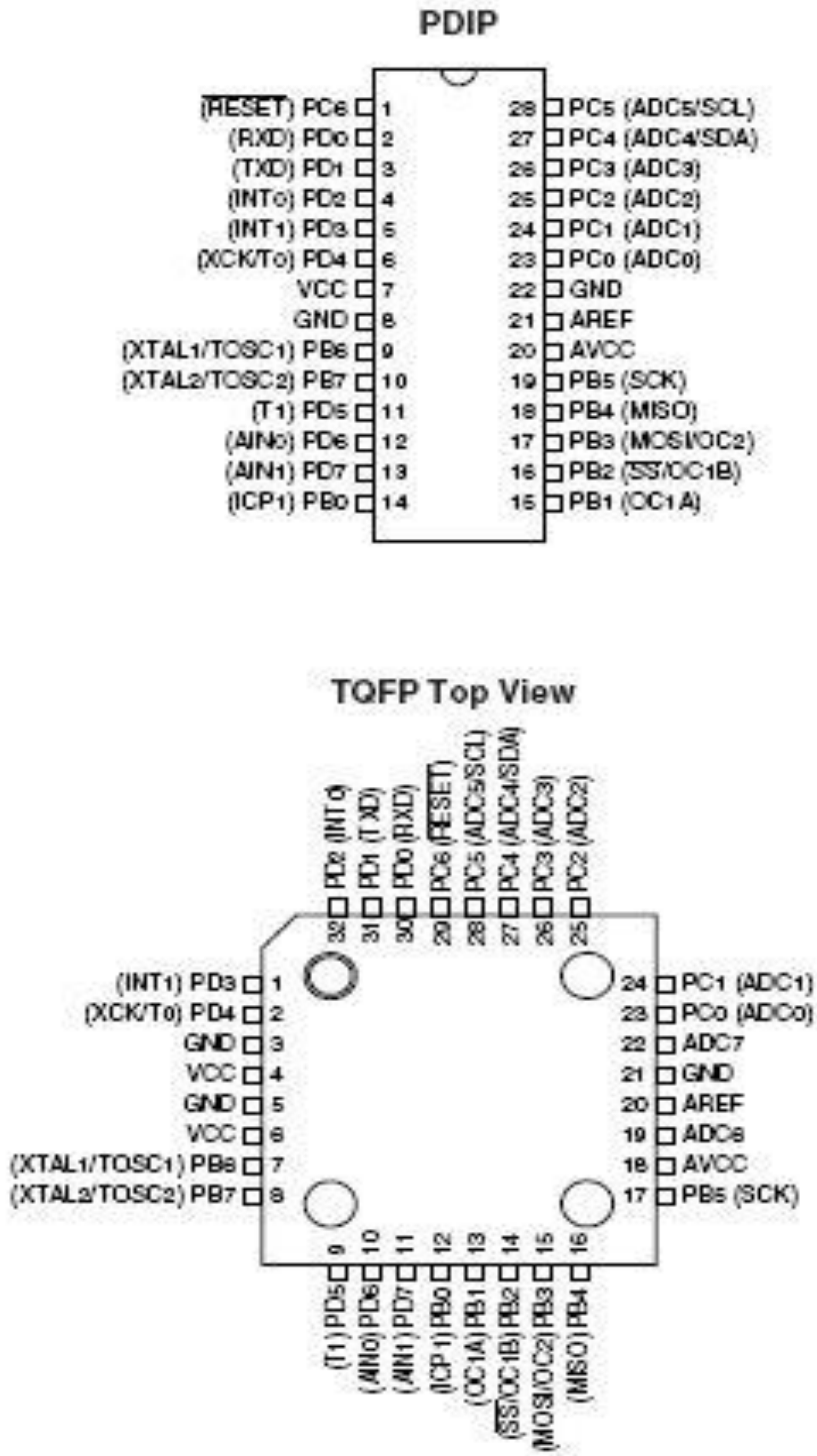

Figure 3.2: Pin Configurations of ATmega8

3.6 Overview of ATmega8 Microcontroller

I/O Ports

23 I/O line can be obtained from three ports; namely Port B, Port C and Port D.

SPI (Serial Peripheral interface):

ATmega8 holds three communication devices integrated. One of them is Serial Peripheral Interface. Four pins are assigned to Atmega8 to implement this scheme of communication.

USART:

One of the most powerful communication solutions is **USART** and ATmega8 supports both synchronous and asynchronous data transfer schemes. It has three pins assigned for that. In many projects, this module is extensively used for PC-Micro controller communication.

TWI (Two Wire Interface):

Another communication device that is present in ATmega8 is Two Wire Interface. It allows designers to set up a commutation between two devices using just two wires along with a common ground connection, As the TWI output is made by means of open collector outputs, thus external pull up resistors are required to make the circuit.

Analog Comparator:

A comparator module is integrated in the IC that provides comparison facility between two voltages connected to the two inputs of the Analog comparator via External pins attached to the micro controller.

Analog to Digital Converter:

The ATmega8 features a 10-bit successive approximation ADC. The ADC is connected to an 8-Channel Analog Multiplexer which allows eight single-ended voltage inputs constructed from the pins of Port C. The single-ended voltage inputs refer to 0V (GND). The ADC contains a Sample and Hold circuit which ensures that the input voltage to the ADC is held at a constant level during conversion. The ADC has a separate analog supply voltage pin, AVCC. AVCC must not differ more than ±0.3V from VCC. Internal reference voltages of nominally 2.56V or AVCC are provided On-chip. The voltage reference may be externally decoupled at the AREF pin by a capacitor for better noise performance.

Memories

The ATmega8 contains 8Kbytes On-chip In-System Reprogrammable Flash memory for program storage. Since all AVR instructions are 16-bits or 32-bits wide, the Flash is organized as 4K × 16 bits. For software security, the Flash Program memory space is divided into two sections, Boot Program section and Application Program section. The Flash memory has an endurance of at least 10,000 write/erase cycles. The ATmega8 Program Counter (PC) is 12 bits wide, thus addressing the 4K Program memory locations.

Internal Clock Source

The Timer/Counter can be clocked directly by the system clock (by setting the CSn2:0 = 1). This provides the fastest operation, with a maximum Timer/Counter clock frequency equal to system clock frequency (fCLK_I/O). Alternatively, one of four taps from the prescalar can be used as a clock source.

Prescalar Reset

The pre scalar is free running (that is, operates independently of the clock select logic of the Timer/Counter) and it is shared by Timer/Counter1 and Timer/Counter0. Since the prescalar is not affected by the Timer/Counter's clock select, the state of the prescaler will have implications for situations where a prescaled clock is used. One example of prescaling artifacts occurs when the timer is enabled and clocked by the prescaler (6 > CSn2:0 > 1). The number of system clock cycles from when the timer is enabled to the first count occurs can be from 1 to N+1 system clock cycles, where N equals the prescaler divisor (8, 64, 256, or 1024).It is possible to use the prescaler reset for synchronizing the Timer/Counter to program execution. However, care must be taken if the other Timer/Counter that shares the same prescaler also uses prescaling.

External Clock Source

An external clock source applied to the T1/T0 pin can be used as Timer/Counter clock (clkT1/clkT0). The T1/T0 pin is sampled once every system clock cycle by the pin synchronization logic. The synchronized (sampled) signal is then passed through the edge detector. Figure 30 shows a functional equivalent block diagram of the T1/T0 synchronization and edge detector logic. The registers are clocked at the positive edge of the internal system clock (clk I/O). The latch is transparent in the high period of the internal system clock. The edge detector generates one clkT1/clkT0 pulse for each positive (CSn2:0 = 7) or negative (CSn2:0 = 6) edge it detects.

Noise Canceler

The noise canceler improves noise immunity by using a simple digital filtering scheme. The noise canceler input is monitored over four samples, and all four must be equal for changing the output that in turn is used by the edge detector. The noise canceler is enabled by setting the *Input Capture Noise Canceler* (ICNC1) bit in *Timer/Counter Control Register B* (TCCR1B). When enabled the noise canceler introduces additional four system clock cycles of delay from a change applied to the input, to the update of the ICR1 Register. The noise canceler uses the system clock and is therefore not affected by the prescaler.

Registers

The Timer/Counter (TCNT2) and Output Compare Register (OCR2) are 8-bit registers. Interrupt request (shorten as Int. Req.) signals are all visible in the Timer Interrupt Flag Register (TIFR). All interrupts are individually masked with the Timer Interrupt Mask Register (TIMSK). TIFR and TIMSK are not shown in the figure since these registers are shared by other timer units. The Timer/Counter can be clocked internally, via the prescaler, or asynchronously clocked from the TOSC1/2 pins, as detailed later in this section.

The asynchronous operation is controlled by the Asynchronous Status Register (ASSR). The Clock Select logic block controls which clock source the Timer/Counter uses to increment (or decrement) its value. The Timer/Counter is inactive when no clock source is selected. The output from the clock select logic is referred to as the timer clock (clkT2).The double buffered Output Compare Register (OCR2) is compared with the Timer/Counter value at all times. The result of the compare can be used by the waveform generator to generate a PWM or variable frequency output on the Output Compare Pin (OC2). The Compare Match event will also set the Compare Flag (OCF2) which can be used to generate an Output Compare interrupt request.

Data Retention

Reliability Qualification results show that the projected data retention failure rate is much less than 1 PPM over 20 years at 85°C or 100 years at 25°C.

General Purpose Register File

The Register File is optimized for the AVR Enhanced RISC instruction set. In order to achieve the required performance and flexibility, the following input/output schemes are supported by the Register File:

- One 8-bit output operand and one 8-bit result input
- Two 8-bit output operands and one 8-bit result input
- Two 8-bit output operands and one 16-bit result input
- One 16-bit output operand and one 16-bit result input

Status Register

The Status Register contains information about the result of the most recently executed arithmetic instruction. This information can be used for altering program flow in order to perform conditional operations. Note that the Status Register is updated after all ALU operations, as specified in the Instruction Set Reference. This will in many cases remove the need for using the dedicated compare instructions, resulting in faster and more compact code. The Status Register is not automatically stored when entering an interrupt routine and restored when returning from an interrupt. This must be handled by software.

Arithmetic Logic Unit – ALU

The high-performance Atmel® AVR ALU operates in direct connection with all the 32 general purpose working registers. Within a single clock cycle, arithmetic operations between general purpose registers or between a register and an immediate are executed. The ALU operations are divided into three main categories – arithmetic, logical, and bit-functions. Some implementations of the architecture also provide a powerful multiplier supporting both signed/unsigned multiplication and fractional format.

Reset and Interrupt Handling

The Atmel® AVR® provides several different interrupt sources. These interrupts and the separate Reset Vector each have a separate Program Vector in the Program memory space. All interrupts are assigned individual enable bits which must be written logic one together with the Global Interrupt Enable bit in the Status Register in order to enable the interrupt. Depending on the Program Counter value, interrupts may be automatically disabled when Boot Lock Bits BLB02 or BLB12 are programmed. This feature improves software security. The lowest addresses in the Program memory space are by default defined as the Reset and Interrupt Vectors. The list also determines the priority levels of the different interrupts. The lower the address the higher is the priority level. RESET has the highest priority, and next is INT0 – the External Interrupt Request 0. The Interrupt Vectors can be moved to the start of the boot Flash section by setting the Interrupt Vector Select (IVSEL) bit in the General Interrupt Control Register. The Reset Vector can also be moved to the start of the boot Flash section by programming the BOOTRST Fuse.

When an interrupt occurs, the Global Interrupt Enable I-bit is cleared and all interrupts are disabled. The user software can write logic one to the I-bit to enable nested interrupts. All enabled interrupts can then interrupt the current interrupt routine. The I-bit is automatically set when a Return from Interrupt instruction – RETI – is executed. There are basically two types of interrupts. The first type is triggered by an event that sets the Interrupt Flag. For these interrupts, the Program Counter is vectored to the actual Interrupt Vector in order to execute the interrupt handling routine, and hardware clears the corresponding Interrupt Flag. Interrupt Flags can also be cleared by writing a logic one to the flag bit position(s) to be cleared. If an interrupt condition occurs while the corresponding interrupt enable bit is cleared, the Interrupt Flag will be set and remembered until the interrupt is enabled, or the flag is cleared by software.

Similarly, if one or more interrupt conditions occur while the global interrupt enable bit is cleared, the corresponding Interrupt Flag(s) will be set and remembered until the global interrupt enable bit is set, and will then be executed by order of priority. The second type of interrupts will trigger as long as the interrupt condition is present. These interrupts do not necessarily have Interrupt Flags. If the interrupt condition disappears before the interrupt is enabled, the interrupt will not be triggered. When the AVR exits from an interrupt, it will always return to the main program and execute one more instruction before any pending interrupt is served. Note that the Status Register is not automatically stored when entering an interrupt routine, nor restored when returning from an interrupt routine. This must be handled by software. When using the CLI instruction to disable interrupts, the interrupts will be immediately disabled. No interrupt will be executed after the CLI instruction, even if it occurs simultaneously with the CLI instruction. The following example shows how this can be used to avoid interrupts during the timed EEPROM write sequence.

Interrupt Response Time

The interrupt execution response for all the enabled Atmel® AVR® interrupts is four clock cycles minimum. After four clock cycles, the Program Vector address for the actual interrupt handling routine is executed. During this 4-clock cycle period, the Program Counter is pushed onto the Stack. The Vector is normally a jump to the interrupt routine, and this jump takes three clock cycles. If an interrupt occurs during execution of a multi-cycle instruction, this instruction is completed before the interrupt is served. If an interrupt occurs when the MCU is in sleep mode, the interrupt execution response time is increased by four clock cycles. This increase comes in addition to the start-up time from the selected sleep mode. A return from an interrupt handling routine takes four clock cycles. During these four clock cycles, the Program Counter (2 bytes) is popped back from the Stack, the Stack Pointer is incremented by 2, and the I-bit in SREG is set.

Stack Pointer

The Stack is mainly used for storing temporary data, for storing local variables and for storing return addresses after interrupts and subroutine calls. The Stack Pointer Register always points to the top of the Stack. Note that the Stack is implemented as growing from higher memory locations to lower memory locations. This implies that a Stack PUSH command decreases the Stack Pointer. The Stack Pointer points to the data SRAM Stack area where the Subroutine and Interrupt Stacks are located. This Stack space in the data

SRAM must be defined by the program before any subroutine calls are executed or interrupts are enabled. The Stack Pointer must be set to point above 0x60. The Stack Pointer is decremented by one when data is pushed onto the Stack with the PUSH instruction, and it is decremented by two when the return address is pushed onto the Stack with subroutine call or interrupt. The Stack Pointer is incremented by one when data is popped from the Stack with the POP instruction, and it is incremented by two when address is popped from the Stack with return from subroutine RET or return from interrupt RETI. The AVR Stack Pointer is implemented as two 8-bit registers in the I/O space. The number of bits actually used is implementation dependent.

Crystal Oscillator

XTAL1 and XTAL2 are input and output, respectively, of an inverting amplifier which can be configured for use as an On-chip Oscillator. Either a quartz crystal or a ceramic resonator may be used. The CKOPT Fuse selects between two different Oscillator amplifier modes. When CKOPT is programmed, the Oscillator output will oscillate a full rail-torail swing on the output. This mode is suitable when operating in a very noisy environment or when the output from XTAL2 drives a second clock buffer. This mode has a wide frequency range. When CKOPT is unprogrammed, the Oscillator has a smaller output swing. This reduces power consumption considerably. This mode has a limited frequency range and it cannot be used to drive other clock buffers. For resonators, the maximum frequency is 8MHz with CKOPT unprogrammed and 16MHz with CKOPT programmed. C1 and C2 should always be equal for both crystals and resonators. The optimal value of the capacitors depends on the crystal or resonator in use, the amount of stray capacitance, and the electromagnetic noise of the environment. Some initial guidelines for choosing capacitors for use with crystals are given in for ceramic resonators, the capacitor values given by the manufacturer should be used.

System Control and Reset

Resetting the AVR during Reset, all I/O Registers are set to their initial values, and the program starts execution from the Reset Vector. If the program never enables an interrupt source, the Interrupt Vectors are not used, and regular program code can be placed at these locations. This is also the case if the Reset Vector is in the Application section while the Interrupt Vectors are in the boot section or vice versa. The I/O ports of the AVR are immediately reset to their initial state when a reset source goes active. This does not require any clock source to be running. After all reset sources have gone inactive, a delay counter is invoked, stretching the internal reset. This allows the power to reach a stable level before normal operation starts. The time-out period of the delay counter is defined by the user through the CKSEL Fuses.

Reset Sources

The ATmega8 has four sources of Reset:

• Power-on Reset. The MCU is reset when the supply voltage is below the Power-on Reset threshold (VPOT).

• External Reset. The MCU is reset when a low level is present on the RESET pin for longer than the minimum pulse length.

• Watchdog Reset. The MCU is reset when the Watchdog Timer period expires and the Watchdog is enabled.

• Brown-out Reset. The MCU is reset when the supply voltage VCC.

3.7 Block Diagram

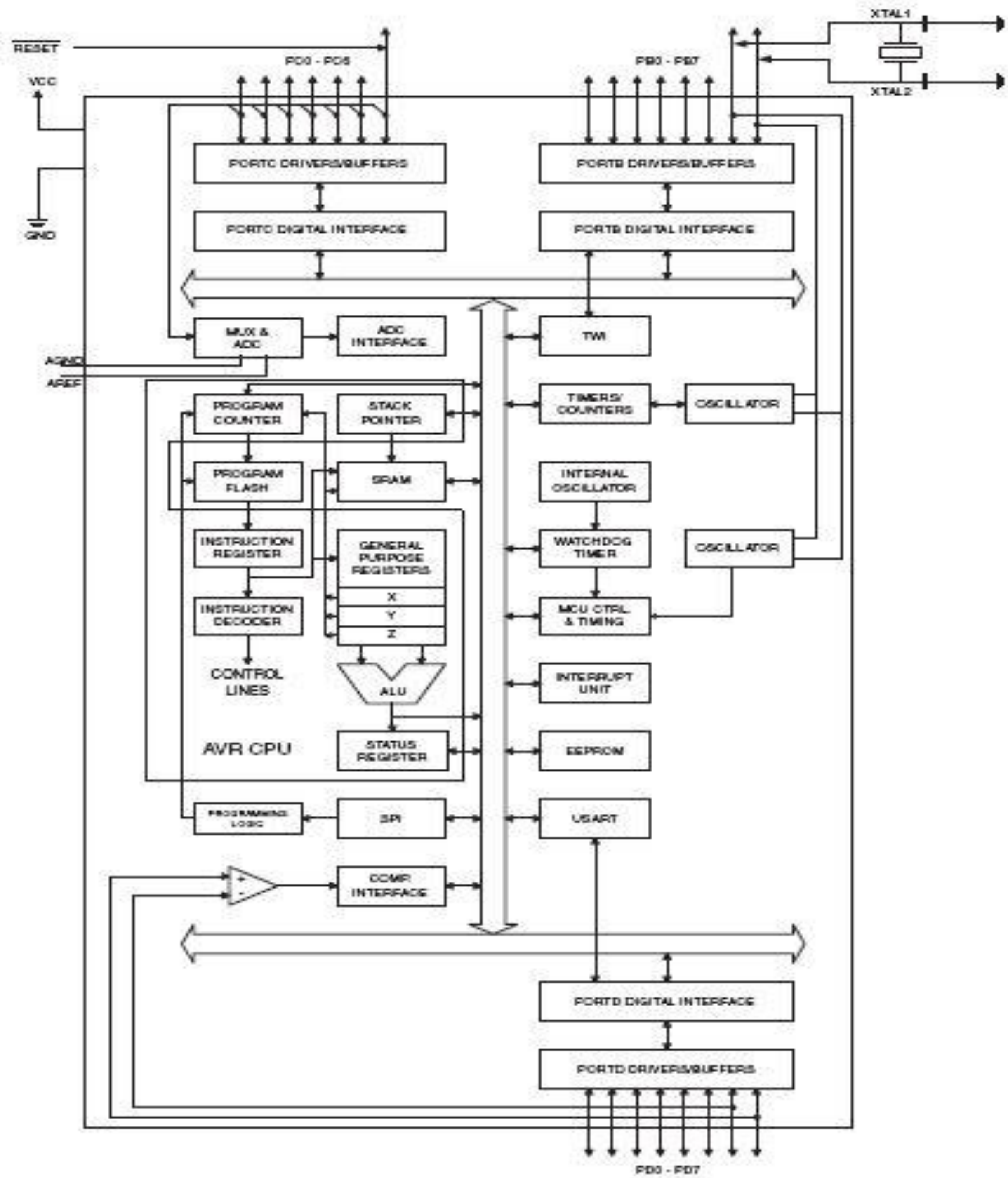

Figure 3.3: Block Diagram of ATmega8

The Atmel® AVR® core combines a rich instruction set with 32 general purpose working registers. All the 32 registers are directly connected to the Arithmetic Logic Unit (ALU), allowing two independent registers to be accessed in one single instruction executed in one clock cycle.

The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers. The ATmega8 provides the following features: 8 Kbytes of In-System Programmable Flash with Read-While-Write capabilities, 512 bytes of EEPROM, 1 Kbyte of SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible Timer/Counters with compare modes, internal and external interrupts, a serial programmable USART, a byte oriented Two wire Serial Interface, a 6-channel ADC (eight channels in TQFP and QFN/MLF packages) with 10-bit accuracy, a programmable Watchdog Timer with Internal Oscillator, an SPI serial port, and five software selectable power saving modes.

The Idle mode stops the CPU while allowing the SRAM, Timer/Counters, SPI port, and interrupt system to continue functioning. The Power down mode saves the register contents but freezes the Oscillator, disabling all other chip functions until the next Interrupt or Hardware Reset. In Power-save mode, the asynchronous timer continues to run, allowing the user to maintain a timer base while the rest of the device is sleeping. The ADC Noise Reduction mode stops the CPU and all I/O modules except asynchronous timer and ADC, to minimize switching noise during ADC conversions.

In Standby mode, the crystal/resonator Oscillator is running while the rest of the device is sleeping. This allows very fast start-up combined with low-power consumption. The device is manufactured using Atmel's high density non-volatile memory technology. The Flash Program memory can be reprogrammed In-System through an SPI serial interface, by a conventional non-volatile memory programmer, or by an On-chip boot program running on the AVR core. The boot program can use any interface to download the application program in the Application Flash memory.

Software in the Boot Flash Section will continue to run while the Application Flash Section is updated, providing true Read-While-Write operation. By combining an 8-bit RISC CPU with In-System Self-Programmable Flash on a monolithic chip, the Atmel ATmega8 is a powerful microcontroller that provides a highly-flexible and cost-effective solution to many embedded control applications. The ATmega8 is supported with a full suite of program and system development tools, including C compilers, macro assemblers, program simulators, and evaluation kits.

3.8 Disclaimer

Typical values contained in this datasheet are based on simulations and characterization of other AVR microcontrollers manufactured on the same process technology. Minimum and Maximum values will be available after the device is characterized.

3.9 Atmel AVR CPU Core

In order to maximize performance and parallelism, the AVR uses a Harvard architecture – with separate memories and buses for program and data. Instructions in the Program memory are executed with a single level pipelining. While one instruction is being executed, the next instruction is pre-fetched from the Program memory. This concept enables instructions to be executed in every clock cycle. The Program memory is In-System Reprogrammable Flash memory. The fast-access Register File contains 32 × 8-bit general purpose working registers with a single clock cycle access time. This allows single-cycle Arithmetic Logic Unit (ALU) operation. In a typical

ALU operation, two operands are output from the Register File, the operation is executed, and the result is stored back in the Register File – in one clock cycle. Six of the 32 registers can be used as three 16-bit indirect address register pointers for Data Space addressing – enabling efficient address calculations. One of these address pointers can also be used as an address pointer for look up tables in Flash Program memory. These added function registers are the 16-bit X-register, Y-register, and Z-register, described later in this section. The ALU supports arithmetic and logic operations between registers or between a constant and a register. Single register operations can also be executed in the ALU. After an arithmetic operation, the Status Register is updated to reflect information about the result of the operation. The Program flow is provided by conditional and unconditional jump and call instructions, able to directly address the whole address space. Most AVR instructions have a single 16-bit word format. Every Program memory address contains a 16-bit or 32-bit instruction. Program Flash memory space is divided in two sections, the Boot program section and the Application program section. Both sections have dedicated Lock Bits for write and read/write protection. The SPM instruction that writes into the Application Flash memory section must reside in the Boot program section. During interrupts and subroutine calls, the return address Program Counter (PC) is stored on the Stack. The Stack is effectively allocated in the general data SRAM, and consequently the Stack size is only limited by the total SRAM size and the usage of the SRAM. All user programs must initialize the SP in the reset routine (before subroutines or interrupts are executed). The Stack Pointer SP is read/write accessible in the I/O space. The data SRAM can easily be accessed through the five different addressing modes supported in the AVR architecture. The memory spaces in the AVR architecture are all linear and regular memory maps. A flexible interrupt module has its control registers in the I/O space with an additional global interrupt enable bit in the Status Register. All interrupts have a separate Interrupt Vector in the Interrupt Vector table. The interrupts have priority in accordance with their Interrupt Vector position. The lower the Interrupt Vector address, the higher the priority. The I/O memory space contains 64 addresses for CPU peripheral functions as Control Registers, SPI, and other I/O functions. The I/O Memory can be accessed directly, or as the Data Space locations following those of the Register File, 0x20 - 0x5F.

3.10 Pin Descriptions

VCC Digital supply voltage.
GND Ground.

Port B (PB7..PB0)XTAL1/XTAL2/TOSC1/TOSC2

Port B is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port B output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port B pins that are externally pulled low will source current if the pull-up resistors are activated. The Port B pins are tri-stated when a reset condition becomes active, even if the clock is not running Depending on the clock selection fuse settings, PB6 can be used as input to the inverting Oscillator amplifier and input to the internal clock operating circuit. Depending on the clock selection fuse settings, PB7 can be used as output from the inverting Oscillator amplifier. If the Internal Calibrated RC Oscillator is used as chip clock source, PB7.6 is used as TOSC2.1input for the Asynchronous Timer/Counter2 if the AS2 bit in ASSR is set.

Port C (PC5..PC0)

Port C is a 7-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port C output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port C pins that are externally pulled low will source current if the pull-up resistors are activated. The Port C pins are tri-stated when a reset condition becomes active, even if the clock is not running.

PC6/RESET

If the RSTDISBL Fuse is programmed, PC6 is used as an I/O pin. Note that the electrical characteristics of PC6 differ from those of the other pins of Port C. If the RSTDISBL Fuse is unprogrammed, PC6 is used as a Reset input. A low level on this pin for longer than the minimum pulse length will generate a Reset, even if the clock is not running. Shorter pulses are not guaranteed to generate a Reset.

Port D (PD7..PD0)

Port D is an 8-bit bi-directional I/O port with internal pull-up resistors (selected for each bit). The Port D output buffers have symmetrical drive characteristics with both high sink and source capability. As inputs, Port D pins that are externally pulled low will source current if the pull-up resistors are activated. The Port D pins are tri-stated when a reset condition becomes active, even if the clock is not running. Port D also serves the functions of various special features of the ATmega8.

RESET

Reset input. A low level on this pin for longer than the minimum pulse length will generate a reset, even if the clock is not running. The minimum pulse length

VCC

VCC is the supply voltage pin for the A/D Converter, Port C (3.0), and ADC (7.6). It should be externally connected to VCC, even if the ADC is not used. If the ADC is used, it should be connected to VCC through a low-pass filter. Note that Port C (5.4) use digital supply voltage, VCC.

AREF

AREF is the analog reference pin for the A/D Converter.

ADC7.6 (TQFP and QFN/MLF Package Only)

In the TQFP and QFN/MLF package, ADC7.6 serve as analog inputs to the A/D converter. These pins are powered from the analog supply and serve as 10-bit ADC channels.

Chapter 4
LCD Display: LM016L

4.1 Description

LCD (Liquid Crystal Display) screen is an electronic display module and find a wide range of applications. A 16x2 LCD display is very basic module and is very commonly used in various devices and circuits. These modules are preferred over seven segments and other multi segment LEDs. The reasons being: LCDs are economical; easily programmable; have no limitation of displaying special & even custom characters (unlike in seven segments), animations and so on.

A 16x2 LCD means it can display 16 characters per line and there are 2 such lines. In this LCD each character is displayed in 5x7 pixel matrix. This LCD has two registers, namely, Command and Data.

The command register stores the command instructions given to the LCD. A command is an instruction given to LCD to do a predefined task like initializing it, clearing its screen, setting the cursor position, controlling display etc. The data register stores the data to be displayed on the LCD. The data is the ASCII value of the character to be displayed on the LCD.

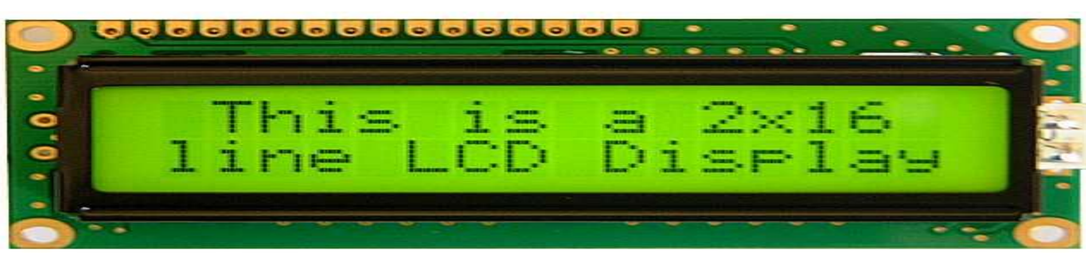

Figure 4.1: LCD Display.

4.2 Pin Diagram.

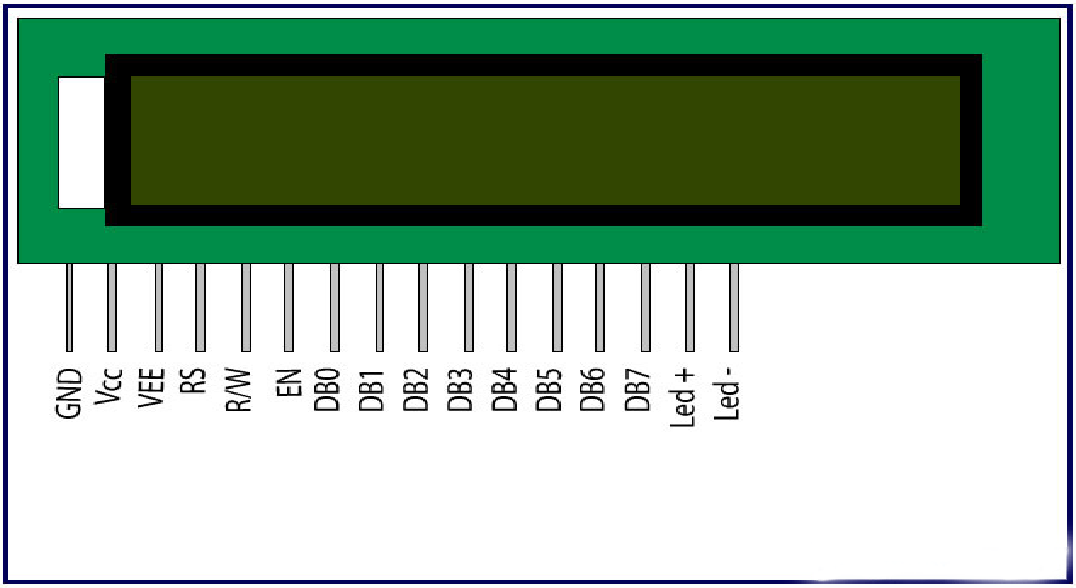

Figure 4.2: Pin Diagram.

4.3 Pin Description

Pin No	Function	Name
1	Ground (0V)	Ground
2	Supply voltage; 5V (4.7V – 5.3V)	Vcc
3	Contrast adjustment; through a variable resistor	V_{EE}

4	**Selects command register when low; and data register when high**	**Register Select**
5	**Low to write to the register; High to read from the register**	**Read/write**
6	**Sends data to data pins when a high to low pulse is given**	**Enable**
7	**8-bit data pins**	**DB0**
8		**DB1**
9		**DB2**
10		**DB3**
11		**DB4**
12		**DB5**
13		**DB6**
14		**DB7**
15	**Backlight V_{CC} (5V)**	**Led+**
16	**Backlight Ground (0V)**	**Led-**

Table 4.1: Pin Description of M016L

Chapter 5
Digtal Thermometer Circuit

5.1 Circuit Diagram

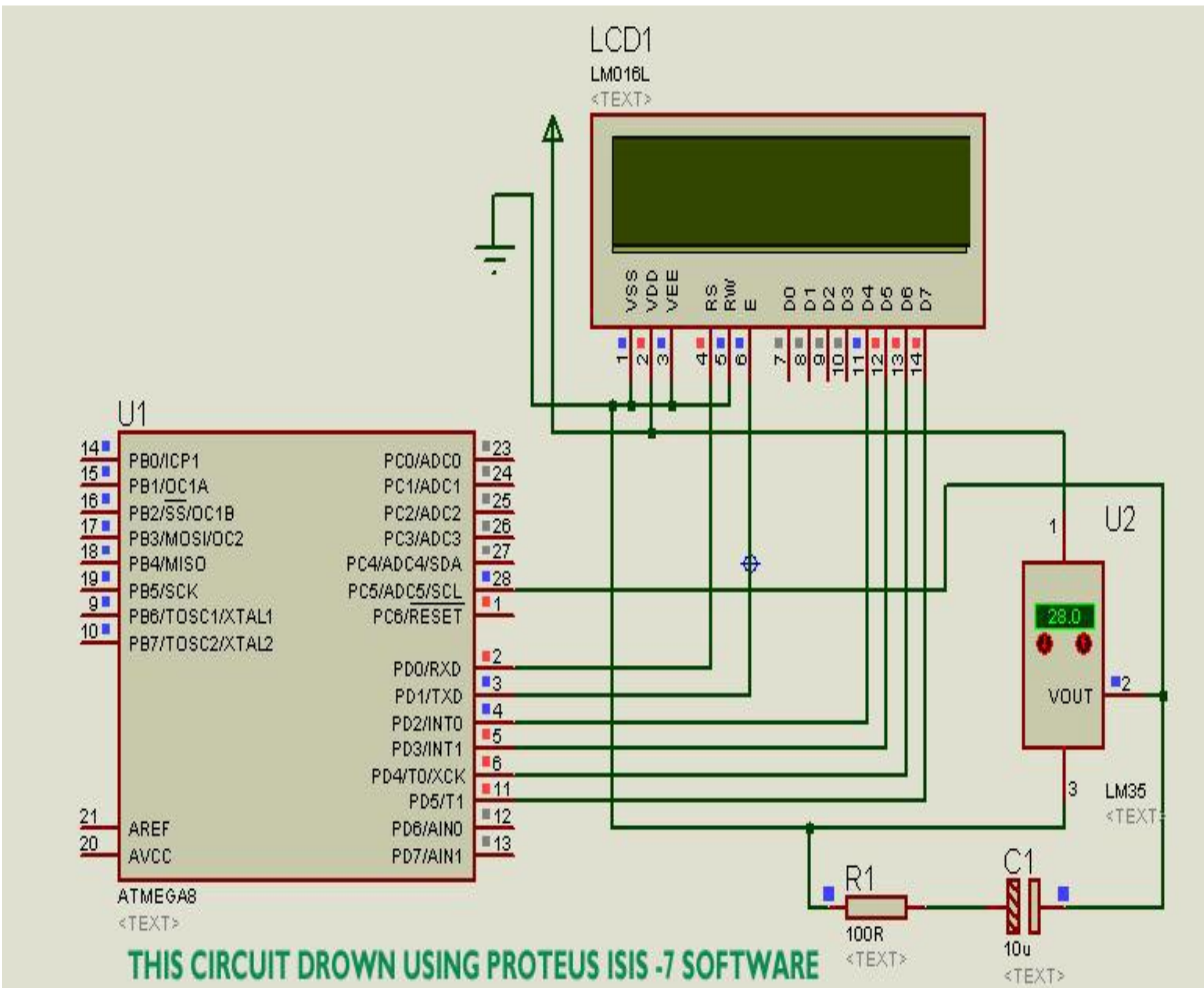

Figure 5.1: Circuit Diagram of Digital Thermometer

5.2 Working Principal

Celsius scale thermometer displays the ambient temperature through a LCD display. It consists of two sections. One is that which senses the temperature. This is a temperature sensor LM 35. The other section converts the temperature value into a suitable number in Celsius scale which is done by the ADC5.

When the system is on, LCD display will show our name and id then the actual temperature of the environment or anybody touched with sensor. Sensor can measure up to 150 degree Celsius. Series connected resistor and capacitor store the output voltage of the sensor. A sensor LM35 is connected to a microcontroller (here atmega8) with pin no. 2. When sensor can sense the temp, it sends an information signal to the microcontroller. Microcontroller takes the signal as an input. A program is written to display the information on convenient format. The output data is displayed through LCD display. LCD display shows the output code of microcontroller in numeric system.

Component

- Transformer.(220v to 5v)
- Resistor.(10k Ω)
- Capacitor.(47 μF)
- Temperature Sensor.(LM 35)
- LCD Display.(LM016L)
- Microcontroller.(ATMEGA8)

Software

- Proteus.(7.7 SP2 ENG v1.0.0)
- AVR Studio .(Version- 4.89)
- Leaper -48 .(Version-10.2)
- AreS7 professional.

5.3 Simulation process

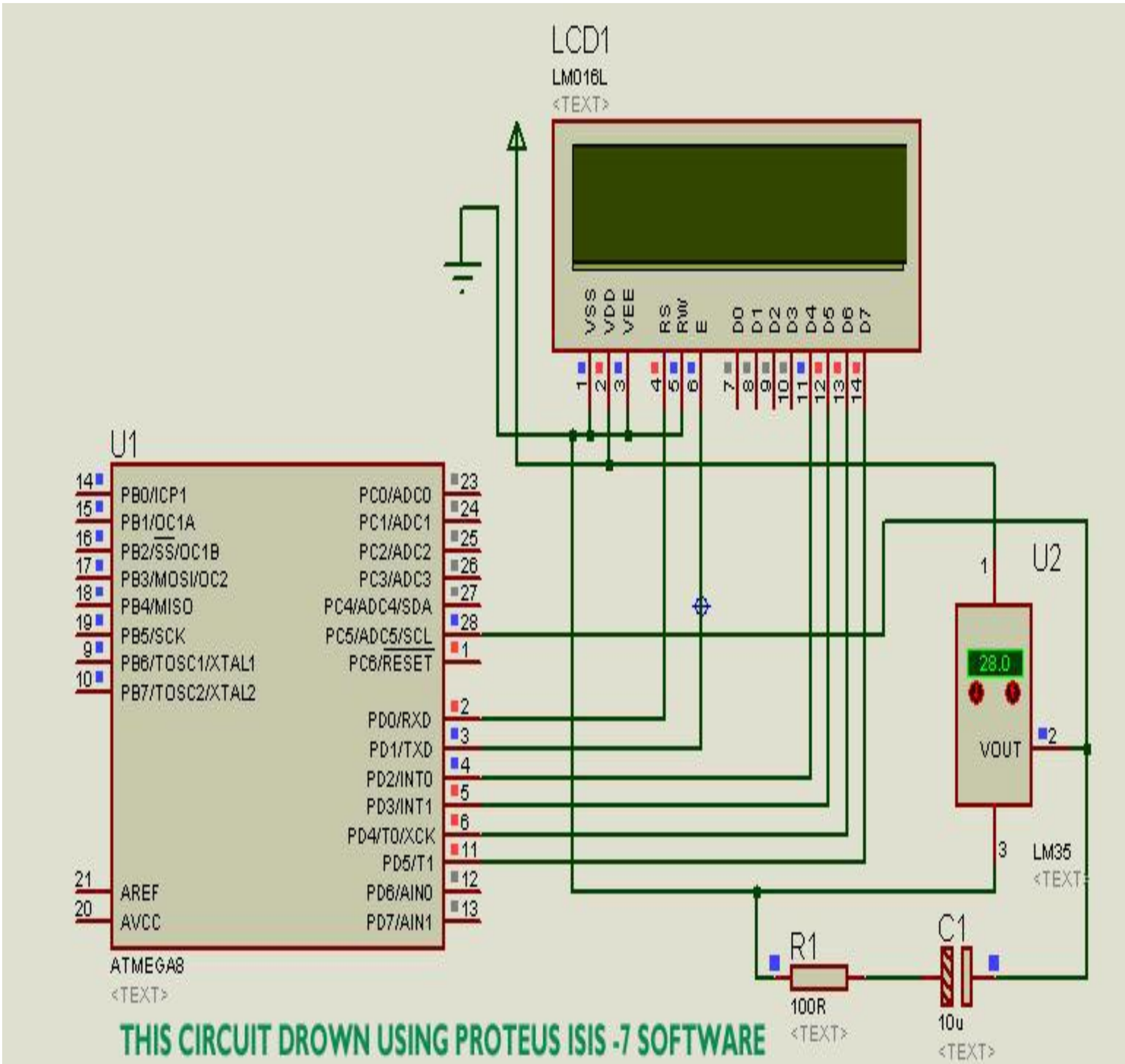

Figure 5.2: Circuit Diagram (Before Simulation)

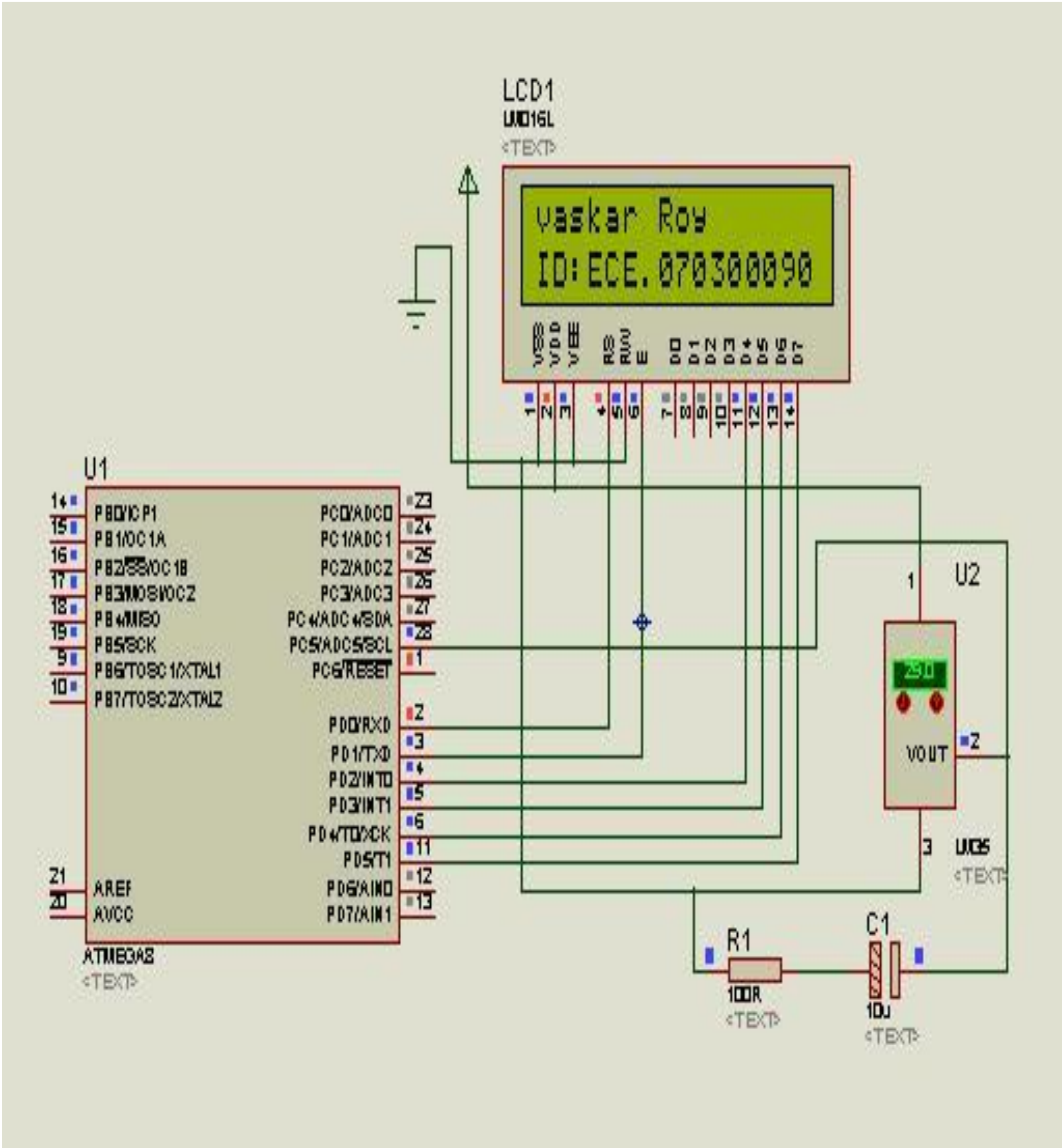

Figure 5.3: Circuit Diagram (After Simulation: step 1)

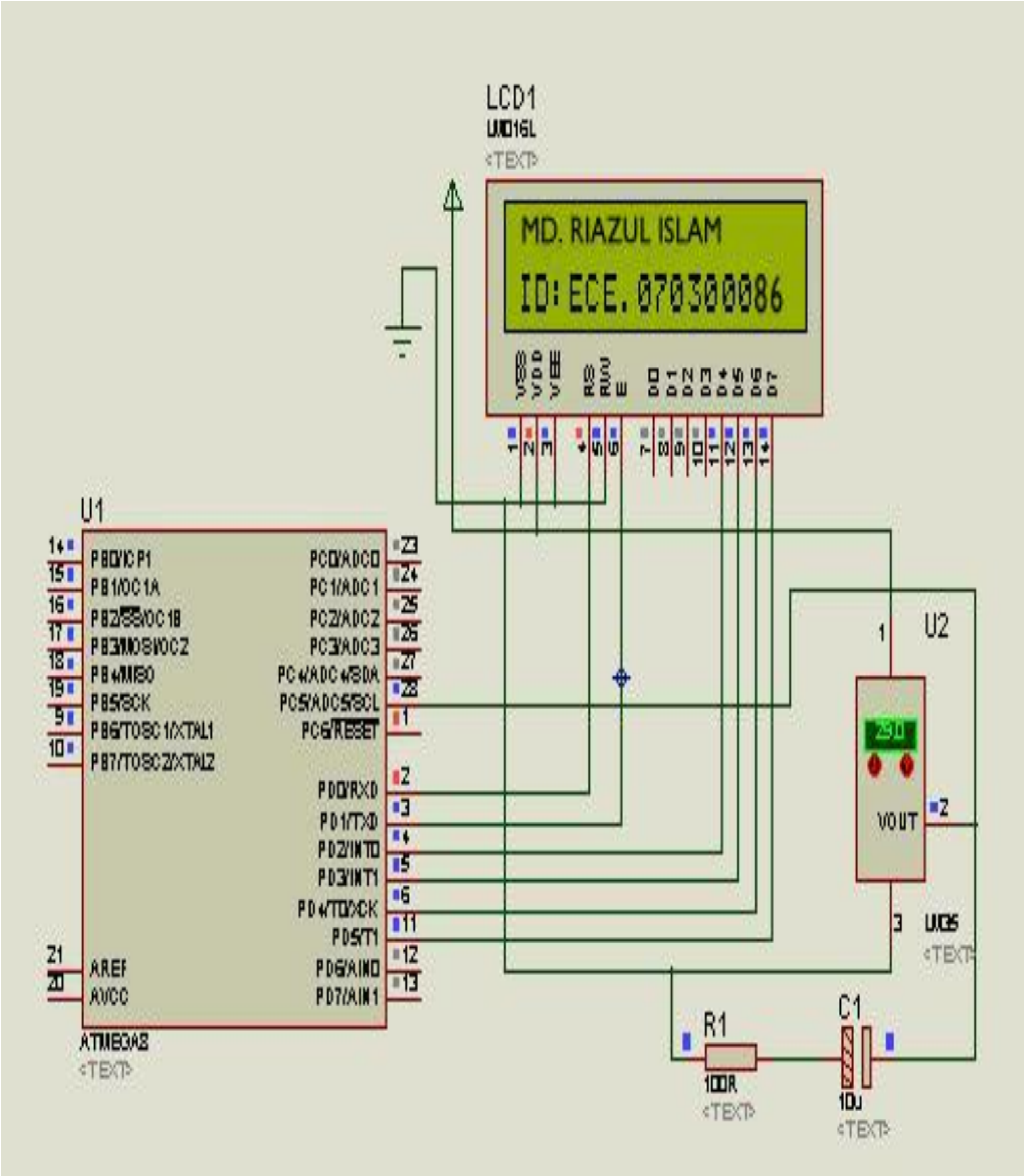

Figure 5.4: Circuit Diagram (After Simulation: step 2).

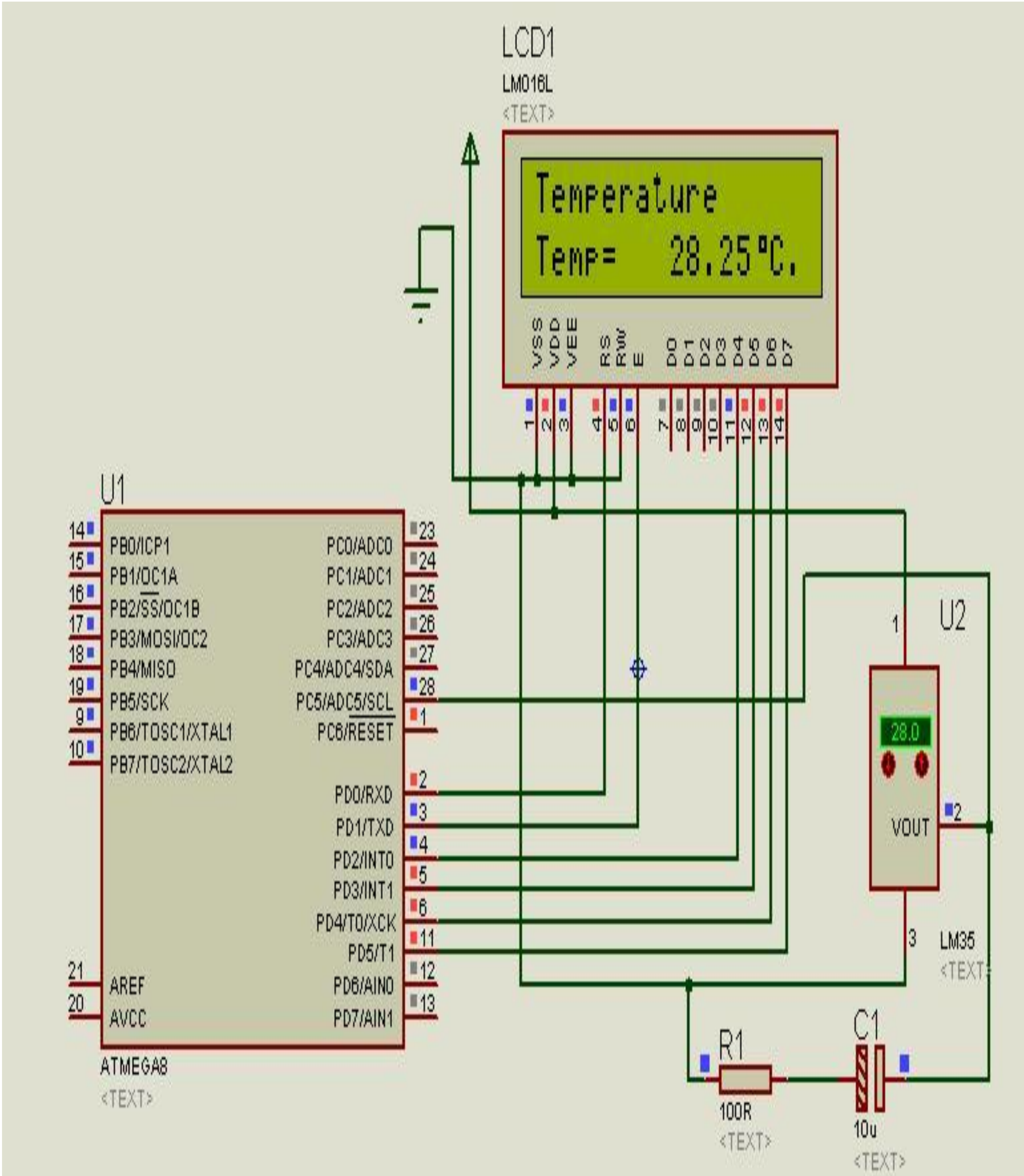

Figure 5.5: Circuit Diagram (After Simulation: step 3)

5.4 PCB Layout

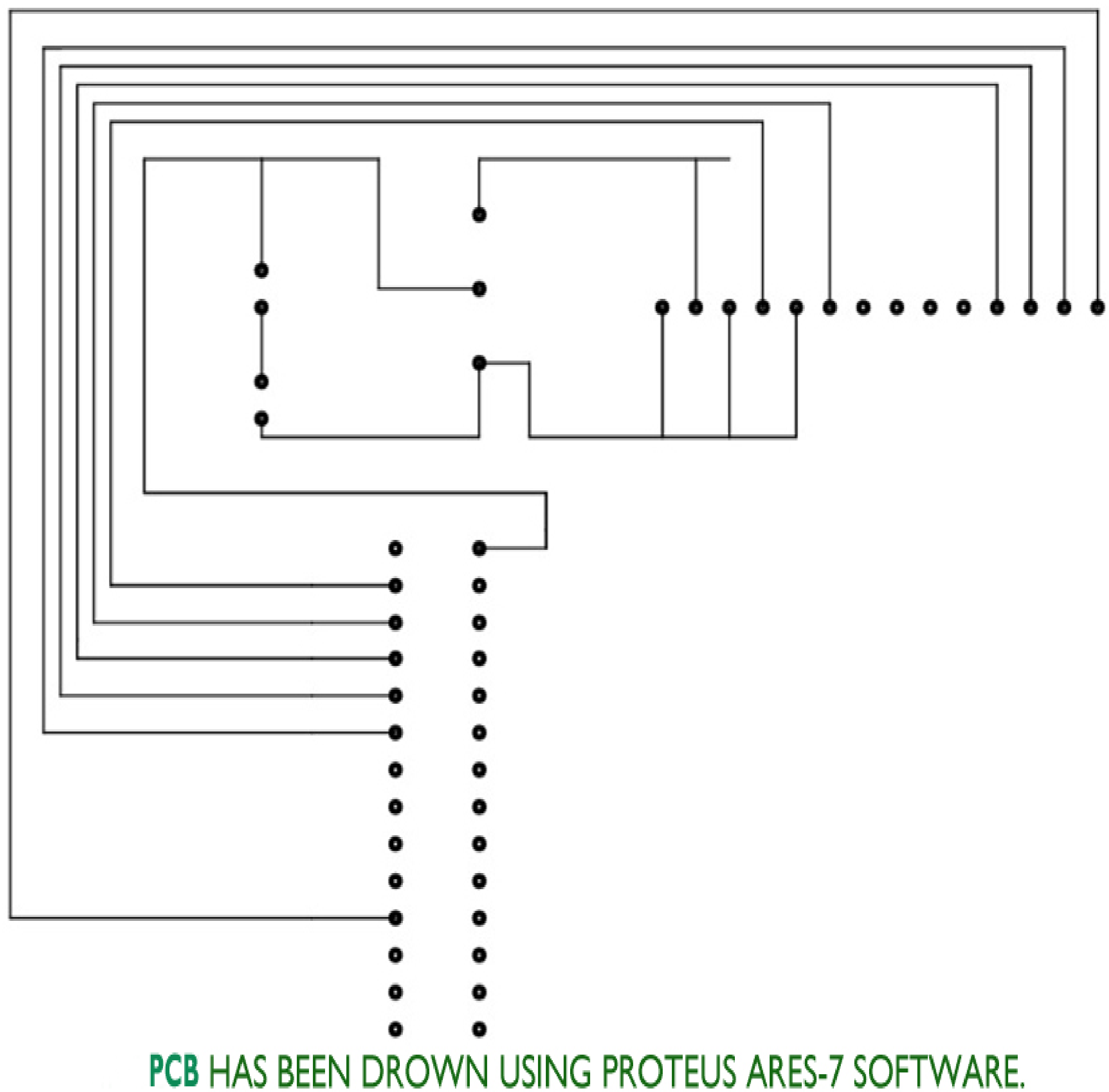

Figure 5.6: PCB Layout

5.5 Different View of Digital Thermometer

Figure 5.7: Front View of Digital Thermometer

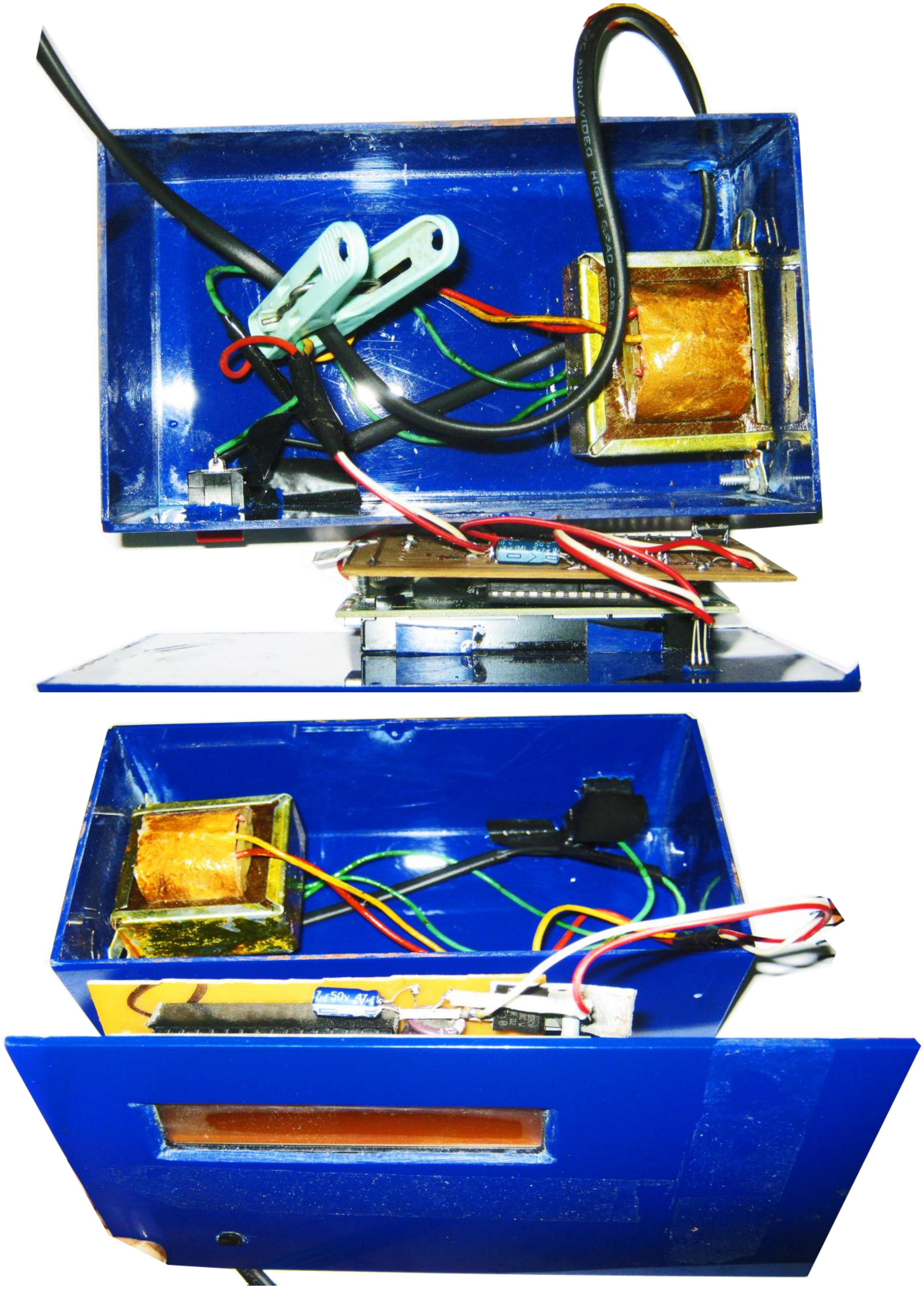

Figure 5.8: Cross-Sectional View of Digital Thermometer

Chapter 6
Conclusion

6.1 Utility

- Measure temperature between -50 °C to 155 °C.
- It can sense every .01°C temperature change and display it.
- The unit (°C) can be replaced with °F that can use to measure viral fever.
- It can response immediately after the temperature change.
- It can also provide a digital input to a computer.

6.2 Limitations

- This device is costly because of using Microprocessor and LCD display.
- Its size is not so comfortable like other digital thermometer.
- Here we use an external power source.
- It is not convenient to measure a body temperature.

6.3 Proposal

- We can use it to measure room temperature of any industry.
- Here we can also measure the humidity of nature and the room.
- By adding a loud speaker we can use it as a fire alarm.
- If the production cost can be reduced that thermometer can be used for daily purpose of general people.

Appendix A

Bascam AVR Code

```
$regfile="m8def.dat"
$crystal=8000000
     Config Portb=Output
     Config Lcd=16*2
     Config Lcdpin=Pin, Rs=Portd.0, E=Portd.1, Db4=Portd.2,
     Db5=portd.3,  Db6=portd.4, Db7=portd.5
     Config Adc=Single, prescaler=Auto, Reference=Internal,
     Deflcdchar 0,7,5,7,32,32,32,32,32
     Dim Adcvalue As Word, Temp As Single,
Start Adc
Cursor Off
Cls
     locate 1,1
          Lcd"vaskar Roy"
     Waitms 100
     locate 2,1
          Lcd "ID:ECE.070300090"
     Waitms 600
Cls
     Locate 1,1
          Lcd "Md.Riazul Islam"
     Waitms 100
     Locate 2,1
          Lcd "ID:ECE.070300086"
     Waitms 600
Cls
     Locate 1,1
          Lcd "Temperature"
     Locate 2,1
          Led"Temp="
     Do
          Adcvalue=Getadc(5)
          Temp=Adcvalue/4
          Locate 2,9
          lcd Temp; Chr(0);"C."
          Waitms 400
     Loop

     End
```

Reference

1. Microcontroller: Internals, Instructions, Programming and Interfacing - Subrata Ghoshal .

2. Microcontroller & Embedded Systems- Mazidi, M. A

3. Microcontroller: Architecture, Programming & Applications- Ayala, Kenneth J.

4. Microcontroller Basics by Davies J H

5. **Microcontroller** by Kenneth Ayala

6. The AVR Microcontroller by Dhananjay Gadre

7. Microcontroller Programming: An Introduction - Syed R. Rizvi

8. “Thermometer “Oxford English Dictionary.

9. T.D. McGee (1988) Principles and Methods of Temperature Measurement ISBN 0-471-62767-4

10.R.S. Doak (2005) Galileo: astronomer and physicist ISBN 0-7565-0813-4 p36

11.T.D. McGee (1988) Principles and Methods of Temperature Measurement page 3, ISBN 0-471-62767-4

12.T.D. McGee (1988) Principles and Methods of Temperature Measurement, pages 2–4 ISBN 0-471-62767-4

13.The Galileo Project: Santorio Santorio

14. R.P. Benedict (1984) Fundamentals of Temperature, Pressure, and Flow Measurements, 3rd ed, ISBN 0-471-89383-8 page 4 R.P. Benedict (1984) Fundamentals of Temperature, Pressure, and Flow Measurements, 3rd ed, ISBN 0-471-89383-8 page 6

15.Sir Thomas Clifford Allbutt, Encyclopædia Britannica

16. Exergen Corporation. Exergen.com. Retrieved on 2011-03-30.

17. Patents By Inventor Francesco Pompei :: Justia Patents. Patents.justia.com. Retrieved on 2011-03-30.

19.Beattie, J.A., Oppenheim, I. (1979). Principles of Thermodynamics, Elsevier Scientific Publishing Company, Amsterdam, ISBN 0-444-41806-7, page 29.

20.Thomsen, J.S. (1962). A restatement of the zeroth law of thermodynamics, Am. J. Phys. **30**: 294-296.

21 Mach, E. (1900). Die Principien der Wärmelehre. Historisch-kritisch entwickelt, Johann Ambrosius Barth, Leipzig, section 22, pages 56-57. English translation edited by McGuinness, B. (1986), Principles of the Theory of Heat, Historically and Critically Elucidated, D. Reidel Publishing, Dordrecht, ISBN 90-277-2206-4, section 5, pp. 48–49, section 22, pages 60–61.

22Truesdell, C.A. (1980). The Tragicomical History of Thermodynamics, 1822-1854, Springer, New York, ISBN 0-387-90403-4.

23 Serrin, J. (1986). Chapter 1, 'An Outline of Thermodynamical Structure', pages 3-32, especially page 6, in New Perspectives in Thermodynamics, edited by J. Serrin, Springer, Berlin, ISBN 3-540-15931-2.

24.Planck, M. (1926). Über die Begründung des zweiten Hauptsatzes der Thermodynamik, S.-B. Preuß. Akad. Wiss. phys. math. Kl.: 453–463.