PROJECT REPORT ON

AUTOMATIC SENSO FAN

Submitted in Partial Fulfillment For Degree

In

Information Technology

By

Rakesh Amrutkar

Under The Guidance of

Prof.Pranav Nerurkar

VEERMATA JIJABAI TECHNOLOGICAL INSTITUTE
MATUNGA, MUMBAI 400051
2018-2019

VEERMATA JIJABAI TECHNOLOGICAL INSTITUTE

(An Autonomous Institute)

H R Mahajani Rd, Matunga, Mumbai, Maharashtra 400019

CERTIFICATE

This is to certify that the project entitled **Automatic Senso Fan** being submitted by **Rakesh Amrutkar** to the department of Information Technology, V.J.T.I , Mumbai for the Award of the Degree in the academic year 2018-19.

Project Guide

Head Of Department

External Examiner

( )

Principal

Date:

Place:

Acknowledgement

This project report is now on compilation just because of following people whom we really thank from the bottom of our heart. We fell fortunate for having **Prof.Pranav Nerurkar** as our project guide.
It was an extremely nice experience to work with **Prof.Pranav Nerurkar** (computer engineering department) as he has done maximum what he can do.
We would also like to thank for the amenities provided by the institute under the authority of our principle.
Our special thanks to all the supporting staff who never left a single stone unturned for providing us with all possible resource,when we are dire need of them.
And last but not the least, we would also like to thank all the individuals who has directly or indirectly helped us in the completion of our project.

Date:

THANK YOU

Abstract

The role of Electricity supply to the Schools and Colleges have been pivotal in the modern era. Devices ranging from fan, lights and many others require electricity to operate on. However turning the devices ON and leaving them ON leads to wastage of energy.This project proposes a system in which the devices like Fan and Light get turned ON only when necessary and turn OFF when not automatically.

The Automatic Light and Fan Switcher Using Weight Sensor Project uses weight sensor to sense the presence of a student on the bench. If a student is detected to be sitting over the bench the Fan and Light will automatically turned ON above the student's bench. Thereafter, when the student gets up and leave the bench the system will turn the Fan and Light OFF again. Also, the system will follow a sequence in turning ON the Fan and Light i.e., if a student sits on the second bench and leaves the first bench vacant, the system will not allow the Fan and Light to get switched ON. But the second bench's Fan and Light would get turned if the first bench was already occupied and the student is then occupying the second bench.

In this way this project smartly controls the Lights and Fans of a classroom and hence makes consumption of energy more efficient and hence saves electricity.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Need

We need this project to save loss of electricity and revolutionize the technology. Since technology has developed so far to protect nature, even this is our little contribution towards it. Thats why we have produced this Automatic Senso-Fan, which does not depend on any other device or human being.

1.2 Information Gathering

first we came up with the idea of this sensor fan. after that we took a little survey to know whether it is really needed or not. we started from our own college where we found out that many students leave the fan and lights on after they leave the classroom which causes a lot loss of electricity. The same happens in the canteen, classrooms, laboratories etc. Not only in our institute but also in other institutes same problem exists. The burden of switching off the fans and lights lies on the peons. Even sometimes peons fail to take this responsibility. We think that our project will help solve this problem. No one will be dependent. Our system provides independent service.

1.3 What Is Embedded System?

An Embedded System is a combination of computer hardware and software, and perhaps additional mechanical or other parts, designed to perform a specific function. An embedded system is a microcontroller-based, software driven, reliable, real-time control system, autonomous, or human or network interactive, operating on diverse physical variables and in diverse environments and sold into a competitive and cost conscious market.
An embedded system is not a computer system that is used primarily for processing, not a software system on PC or UNIX, not a traditional business or scientific application. High-end embedded lower end embedded systems. High-end embedded system - Generally 32, 64 Bit Controllers used with OS. Examples Personal Digital Assistant and Mobile phones etc .Lower end embedded systems - Generally 8,16 Bit Controllers used with an minimal operating systems and hardware layout designed for the specific purpose.

1.3.1 System Design Cell

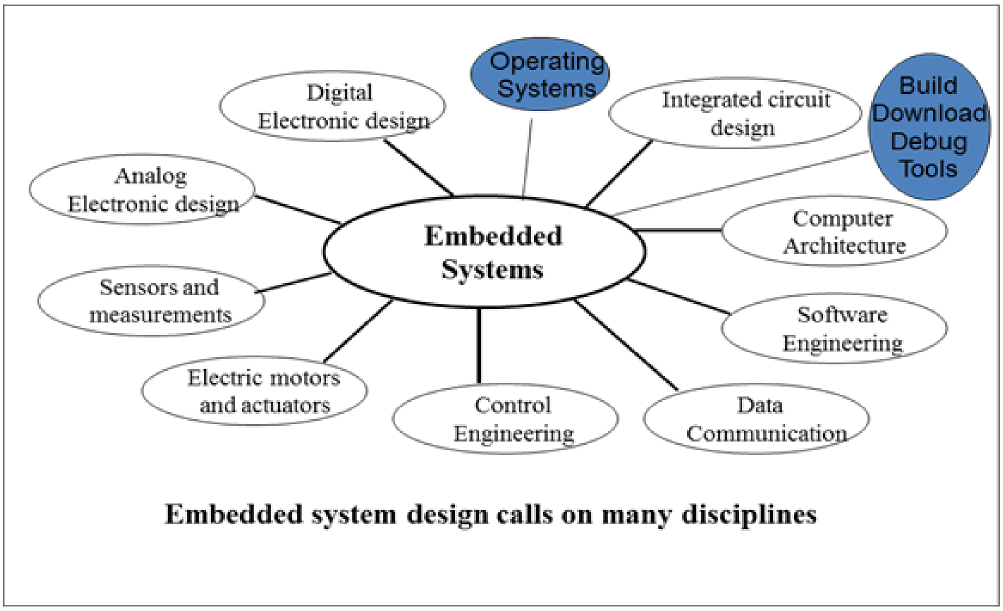


Figure 1.1: Embedded system design calls

1.3.2 Embedded System Design Cycle

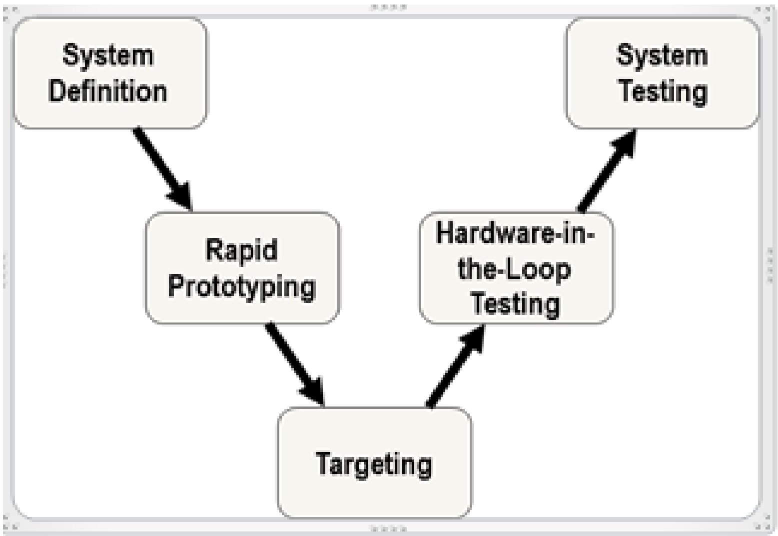


Figure 1.2: V Diagram

1.4 Characteristics Of Embedded System

- An embedded system is any computer system hidden inside a product other than a computer.
- They will encounter a number of difficulties when writing embedded system software in addition to those we encounter when we write applications
  - Throughput Our system may need to handle a lot of data in a short period of time.
  - ResponseOur system may need to react to events quickly TestabilitySetting up equipment to test embedded software can be difficult
  - DebugabilityWithout a screen or a keyboard, finding out what the software is doing wrong (other than not working) is a troublesome problem
  - Reliability embedded systems must be able to handle any situation without human intervention
  - Memory space Memory is limited on embedded systems, and you must make the software and the data fit into whatever memory exists. Program installation you will need special tools to get your software into embedded systems.
  - Power consumption Portable systems must run on battery power, and the software in these systems must conserve power.
  - Processor hogs computing that requires large amounts of CPU time can complicate the response problem.
  - Cost Reducing the cost of the hardware is a concern in many embedded system projects; software often operates on hardware that is barely adequate for the job.
- Embedded systems have a microprocessor/ microcontroller and a memory. Some have a serial port or a network connection. They usually do not have keyboards, screens or disk drives.

1.5 Application

- Military and aerospace embedded software applications
- Communication Applications
- Industrial automation and process control software
- Mastering the complexity of applications.
- Reduction of product design time.
- Real time processing of ever increasing amounts of data.
- Intelligent, autonomous sensors.

1.6 Classification

- Real Time Systems.
- RTS is one which has to respond to events within a specified deadline.
- A right answer after the dead line is a wrong answer.

1.6.1 RTS Classification

- Hard Real Time Systems
- Soft Real Time System

Hard Real Time System

- "Hard" real-time systems have very narrow response time.
- Example: Nuclear power system, Cardiac pacemaker.

Soft Real Time System

- "Soft" real-time systems have reduced constrains on "lateness" but still must operate very quickly and repeatable.
- Example: Railway reservation system takes a few extra seconds the data remains valid.

Chapter 2

Block Diagram

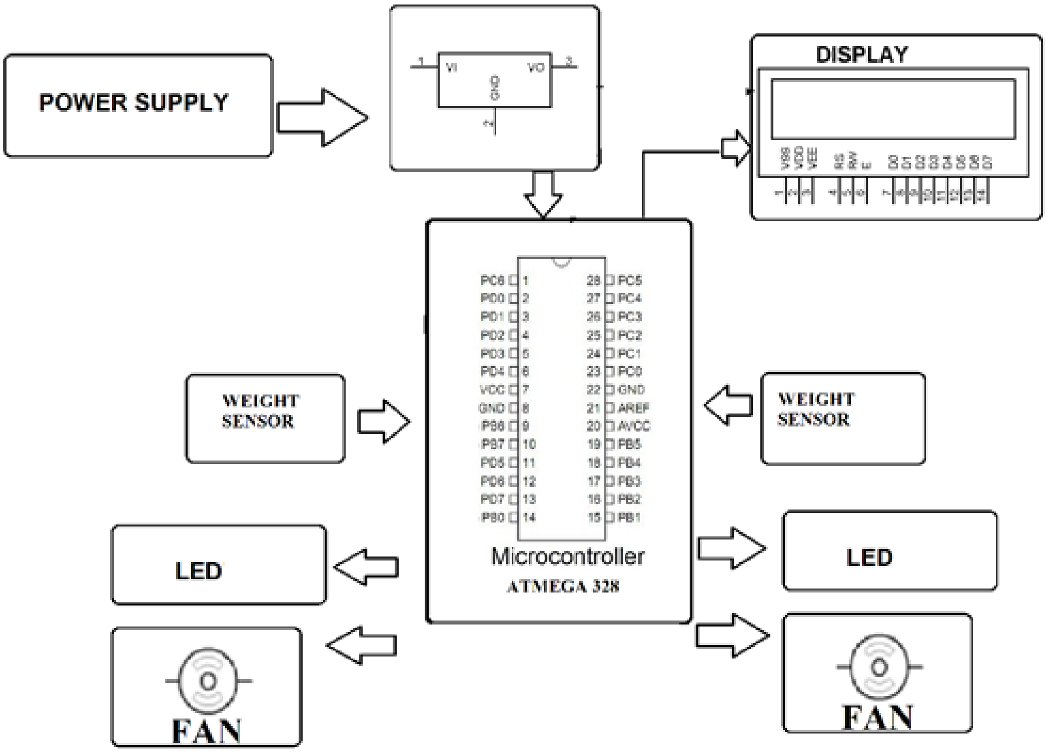


Figure 2.1: Project block diagram

Chapter 3

Hardware Requirements

Hardware Component

1. TRANSFORMER (230 12 V AC)
2. VOLTAGE REGULATOR (LM 7805)
3. RECTIFIER
4. FILTER
5. ATMEGA 328 Microcontroller
6. ULN2003
7. RELAY
8. LCD
9. LED
10. 1N4007
11. RESISTOR
12. CAPACITOR

3.1 Transformer

Transformers convert AC electricity from one voltage to another with a little loss of power. Step-up transformers increase voltage, step-down transformers reduce voltage.Most power supplies use a step-down transformer to reduce the dangerously high voltage to a safer low voltage.

Figure 3.1: A typical tranformer

The input coil is called the primary and the output coil is called the secondary. There is no electrical connection between the two coils; instead they are linked by an alternating magnetic field created in the soft-iron core of the transformer. The two lines in the middle of the circuit symbol represent the core. Transformers waste very little power so the power out is (almost) equal to the power in. Note that as voltage is stepped down and current is stepped up.
The ratio of the number of turns on each coil, called the turns ratio, determines the ratio of the voltages. A step-down transformer has a large number of turns on its primary (input) coil which is connected to the high voltage mains supply, and a small number of turns on its secondary (output) coil to give a low output voltage.
TURNS RATIO = (Vp / Vs) = (Np / Ns)
Where,
Vp = primary (input) voltage
Vs = secondary (output) voltage
Np = number of turns on primary coil
Ns = number of turns on secondary coil
Ip = primary (input) current
Is = secondary (output) current.

Ideal Power Equation

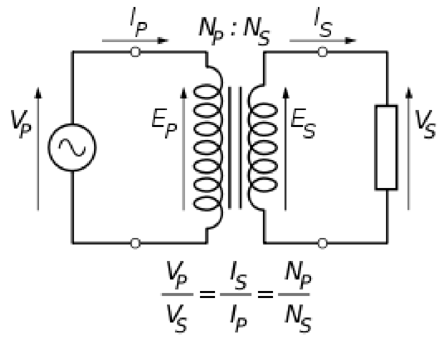


Figure 3.2: Ideal transformer circuit

The ideal transformer as a circuit element.If the secondary coil is attached to a load that allows current to flow, electrical power is transmitted from the primary circuit to the secondary circuit. Ideally, the transformer is perfectly efficient; all the incoming energy is transformed from the primary circuit to the magnetic field and into the secondary circuit.If this condition is met, the incoming electric power must equal the outgoing power:
Giving the ideal transformer equation.Transformers normally have high efficiency, so this formula is a reasonable approximation.If the voltage is increased, then the current is decreased by the same factor. The impedance in one circuit is transformed by the square of the turns ratio. For example, if an impedance Zs is attached across the terminals of the secondary coil, it appears to the primary circuit to have an impedance of (Np/Ns)2 Zs. This relationship is reciprocal, so that the impedance Zp of the primary circuit appears to the secondary to be (Ns/Np)2 Zp.

3.2 Voltage Regulator 7805

Features

- Output Current up to 1A. Voltages of 5, 6, 8, 9, 10, 12, 15, 18, 24V.
- Thermal Overload Protection.
- Short Circuit Protection.
- Output Transistor Safe Operating Area Protection.

Description

The LM78XX/LM78XXA series of three-terminal positive regulators are available in the TO-220/D-PAK package and with several fixed output voltages, making them useful in a Wide range of applications. Each type employs internal current limiting, thermal shutdown and safe operating area protection, making it essentially indestructible. If adequate heat sinking is provided, they can deliver over 1A output Current. Although designed primarily as fixed voltage regulators, these devices can be used with external components to obtain adjustable voltages and currents.

Internal Block Diagram

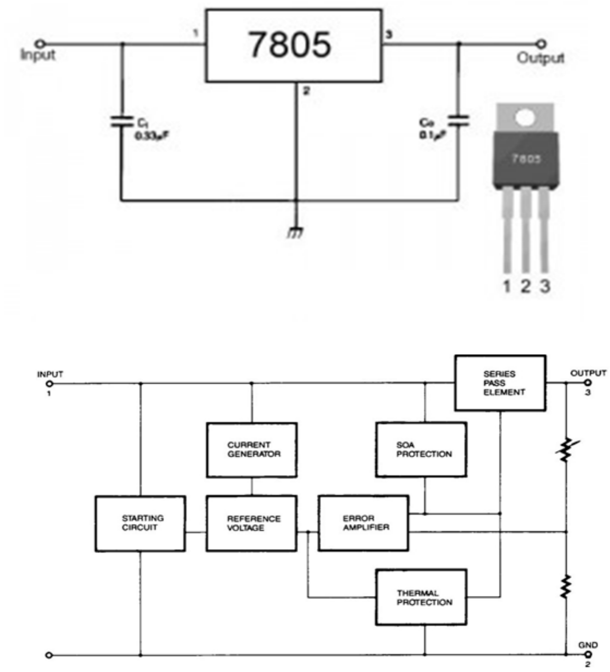


Figure 3.3: Voltage Regulator

3.3 Rectifier

A rectifier is an electrical device that converts alternating current (AC), which periodically reverses direction, to direct current (DC), current that flows in only one direction, a process known as rectification. Rectifiers have many uses including as components of power supplies and as detectors of radio signals. Rectifiers may be made of solid state diodes, vacuum tube diodes, mercury arc valves, and other components. The output from the transformer is fed to the rectifier. It converts A.C. into pulsating D.C.
The rectifier may be a half wave or a full wave rectifier. In this project, a bridge rectifier is used because of its merits like good stability and full wave rectification. In positive half cycle only two diodes (1 set of parallel diodes) will conduct, in negative half cycle remaining two diodes will conduct and they will conduct only in forward bias only.

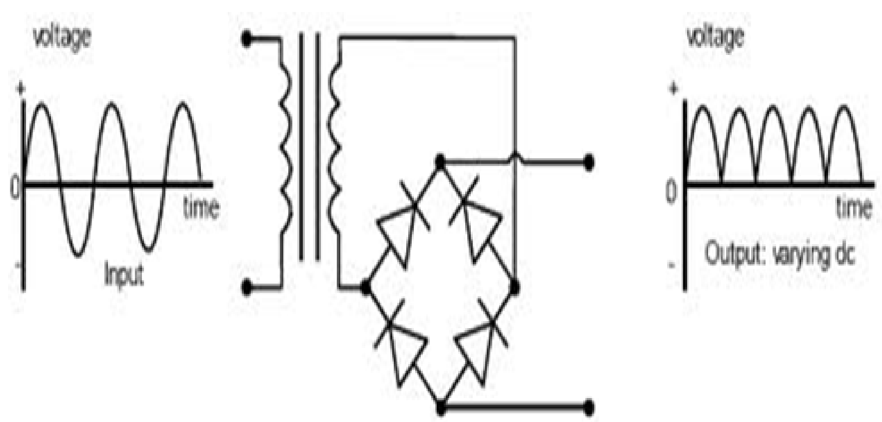


Figure 3.4: Circuit diagram of rectifier

3.4 Filter

Capacitive filter is used in this project. It removes the ripples from the output of rectifier and smoothness the D.C. Output received from this filter is constant until the mains voltage and load is maintained constant. However, if either of the two is varied, D.C. voltage received at this point changes. Therefore a regulator is applied at the output stage.
The simple capacitor filter is the most basic type of power supply filter. The use of this

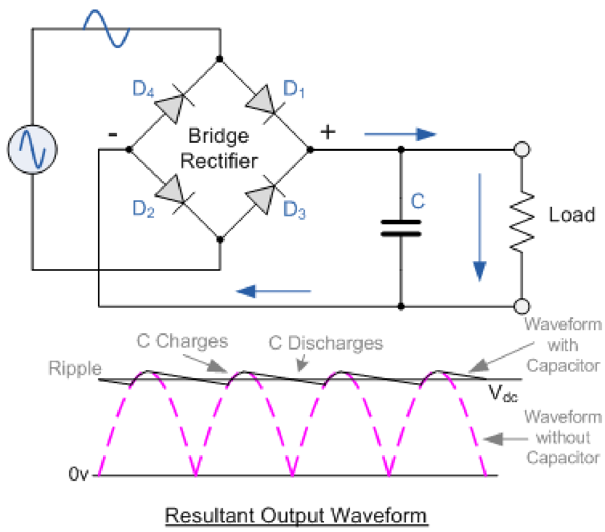


Figure 3.5: Circuit Diagram of Filter

filter is very limited. It is sometimes used on extremely high-voltage, low-current power supplies for cathode-ray and similar electron tubes that require very little load current from the supply. This filter is also used in circuits where the power-supply ripple frequency is not critical and can be relatively high. Below figure can show how the capacitor charges and discharges.

3.5 Atmega328

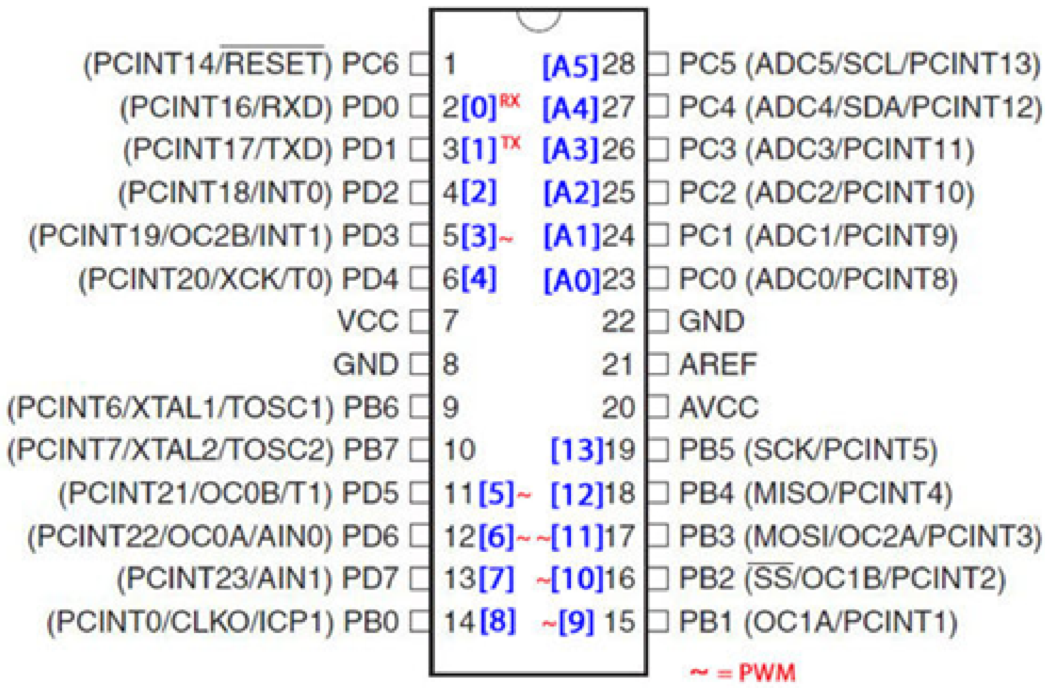


Figure 3.6: Pin diagram of ATMEGA328

3.5.1 Introduction

The Atmel ATmega328P is a 32K 8-bit microcontroller based on the AVR architecture. Many instructions are executed in a single clock cycle providing a throughput of almost 20 MIPS at 20MHz. The ATMEGA328-PU comes in an PDIP 28 pin package and is suitable for use on our 28 pin AVR Development Board.

The computer on one hand is designed to perform all the general purpose tasks on a single machine like you can use a computer to run a software to perform calculations or you can use a computer to store some multimedia file or to access internet through the browser, whereas the microcontrollers are meant to perform only the specific tasks, for e.g., switching the AC off automatically when room temperature drops to a certain defined limit and again turning it ON when temperature rises above the defined limit.

There are number of popular families of microcontrollers which are used in different applications as per their capability and feasibility to perform the desired task, most common of these are 8051, AVR and PIC microcontrollers. In this we will introduce you with AVR family of microcontrollers.

3.5.2 Features Include

- High Performance, Low Power Design
- 8-Bit Microcontroller Atmel AVR advanced RISC architecture
  - 131 Instructions most of which are executed in a single clock cycle

  - Up to 20 MIPS throughput at 20 MHz
  - 32 x 8 working registers
  - 2 cycle multiplier
- Memory Includes
  - 32KB of of programmable FLASH
  - 1KB of EEPROM
  - 2KB SRAM
  - 10,000 Write and Erase Cycles for Flash and 100,000 for EEPROM
  - Data retention for 20 years at 85C and 100 years at 25C
  - Optional boot loader with lock bits
    * In System Programming (ISP) by via boot loader True Read-While-Write operation
  - Programming lock available for software security
- Features Includes
  - 2 x 8-bit Timers/Counters each with independent prescaler and compare modes
  - A single 16-bit Timer/Counter with an independent prescaler, compare and capture modes
  - Real time counter with independent oscillator
  - 10 bit, 6 channel analog to digital Converter
  - 6 pulse width modulation channels
  - Internal temperature sensor
  - Serial USART (Programmable)
  - Master/Slave SPI Serial Interface - (Philips I2C compatible)
  - Programmable watchdog timer with independent internal oscillator
  - Internal analog comparator
  - Interrupt and wake up on pin change
- Additional Features Features
  - Internal calibrated oscillator
  - Power on reset and programmable brown out detection
  - External and internal interrupts
  - 6 sleep modes including idle, ADC noise reduction, power save, power down, standby, and extended standby
- I/O and Package
  - 23 programmable I/O lines
  - 28 pin PDIP package
- Operating voltage

  - 1.8 - 5.5V
- Operating temperature range
  - 40C to 85C
- Speed Gradese
  - 0-4 MHz at 1.8-5.5V
  - 0-10 MHz at 2.7-5.5V
  - 0-20 MHz at 4.5-5.5V
- Low power consumption mode at 1.8V, 1 MHz and 25C
- Active Mode: 0.3 mA
- Power-down Mode: 0.1 A
- Power-save Mode: 0.8 A (Including 32 kHz RTC)

Flash: 32 Kbytes
EEPROM: 1 Kbytes
SRAM: 2 Kbytes
Max I/O Pins: 23
Frequency Max: 20 MHz
VCC: 1.8-5.5
10-bit A/D Channels: 6
Analog Comparator: Yes
16-bit Timers: 1
8-bit Timer: 2
Brown Out Detector: Yes
Ext Interrupts: 2
Hardware Multiplier: Yes
Interrupts: 26
ISP: Yes
On Chip Oscillator: Yes
PWM Channels: 6
RTC: Yes
Self Program Memory: Yes
SPI: 1
TWI: Yes
UART: 1
Watchdog: Yes
Package: Lead Free PDIP 28

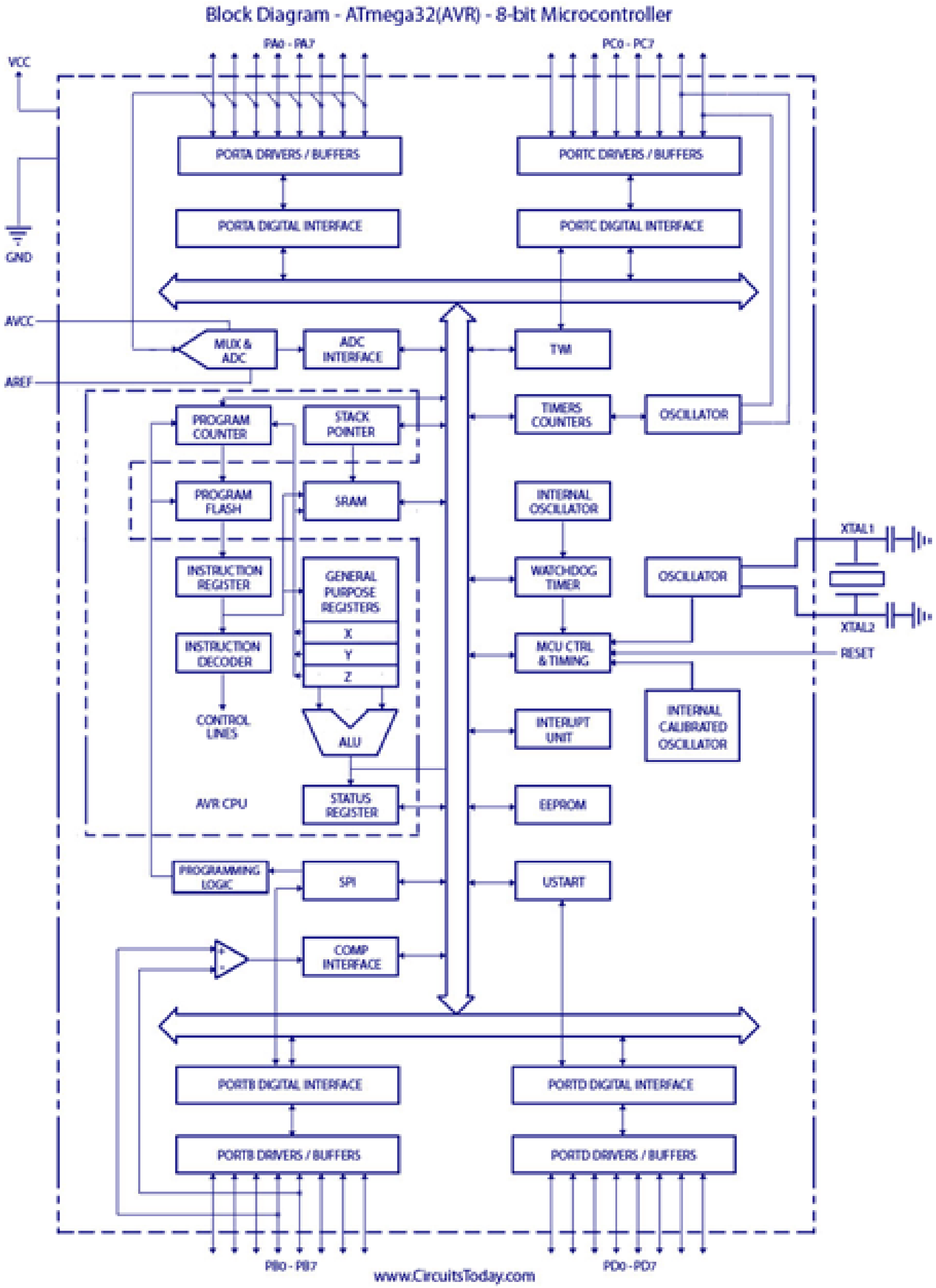


Figure 3.7: Block daigram of atmega 328

3.6 ULN2003

3.6.1 Relay Driver

ULN2003 is a high voltage and high current Darlington transistor array.

3.6.2 Description

The ULN2003 is a monolithic high voltage and high current Darlington transistor arrays. It consists of seven NPN Darlington pairs that feature high-voltage outputs with common-cathode Clamp diode for switching inductive loads. The collector-current rating of a single Darlington pair is 500mA. The Darlington pairs may be paralleled for higher current capability. Applications include relay drivers, hammer drivers, lamp drivers, display drivers (LED gas discharge), line drivers, and logic buffers.
The ULN2003 has a 2.7kW series base resistor for each Darlington pair for operation directly with TTL or 5V CMOS devices.

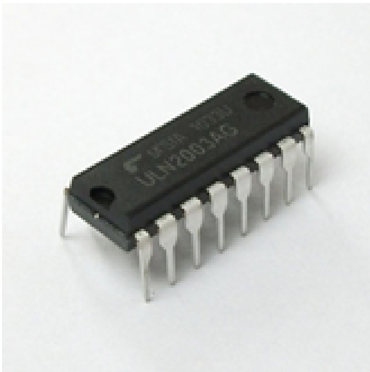


Figure 3.8: Pin Diagram

3.6.3 Features

- Pin no.:16
- Temperature, Operating Range:-20C to +85C
- Transistor Polarity: NPN
- Transistors, No. of:7
- Case Style:DIP-16
- Temp, Op. Min:-20C
- Temp, Op. Max:85C
- Base Number:2003
- Channels, No. of:7
- Current, Output Max:500mA

3.6.4 Pin Diagram

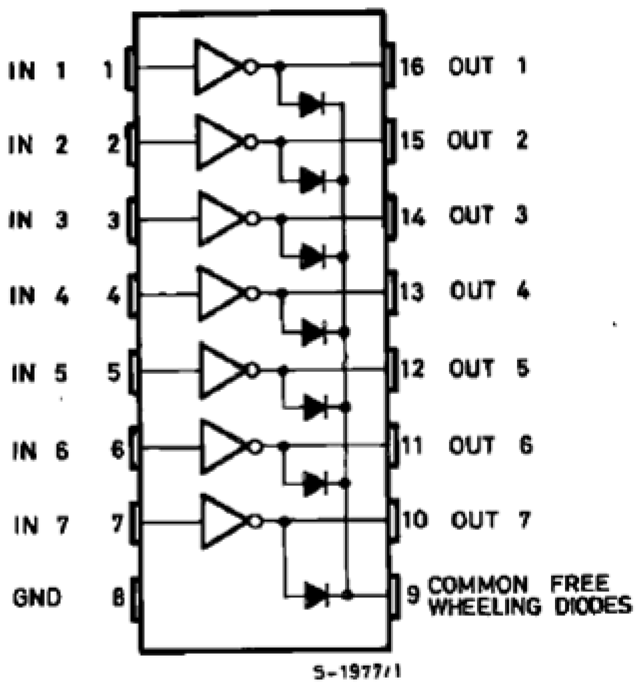


Figure 3.9: Pin Diagram

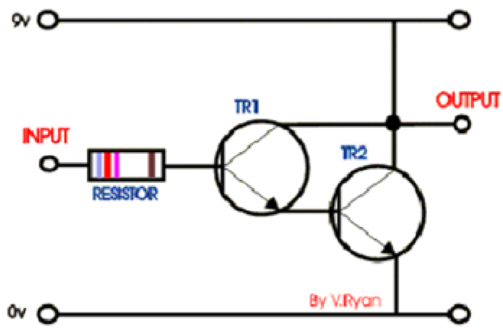


Figure 3.10: Darlington's Pairs

Darlington pairs are back to back connection of two transistors with some source resistors and when they are arranged as shown in the circuit they are used to amplify weak signals. The amount by which the weak signal is amplified is called the GAIN. .

3.6.5 Features Of Driver

- Seven Darlingtons per package
- Output currents500mA per driver(600mA peak)
- Integrated suppression diodes for inductive loads

- Outputs can be paralleled for high currents
- TTL/CMOS/PMOS/DTL compatible inputs.
- Inputs pinned opposite to outputs
- Simplified layout

Figure shows the Darlington pair connection of transistor. The circuit above is a Darlington Pair driver. The first transistors emitter feeds into the second transistors base and as a result the input signal is amplified by the time it reaches the output. The important point to remember is that the Darlington Pair is made up of two transistors.

Features

- 500mA rated collector current (Single output).
- High-voltage outputs: 50V.
- Inputs compatible with various types of logic.
- Relay driver application.

3.7 Relay

A relay is an electrically operated switch. Many relays use an electromagnet to operate a switching mechanism mechanically, but other operating principles are also used. Relays are used where it is necessary to control a circuit by a low-power signal (with complete electrical isolation between control and controlled circuits), or where several circuits must be controlled by one signal.
A relay is an electrically operated switch. Current flowing through the coil of the relay creates

Figure 3.11: Types of relay

a magnetic field which attracts a lever and changes the switch contacts. The coil current can be on or off so relays have two switch positions and most have double throw (changeover) switch contacts as shown in the diagram.

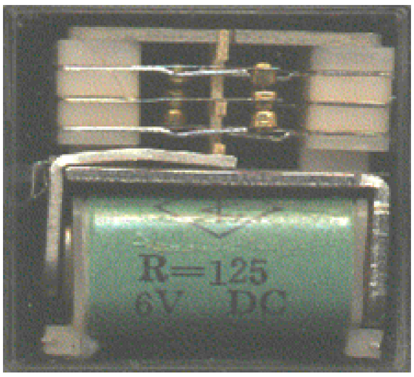


Figure 3.12: Relay showing coil and switch contacts

Relays allow one circuit to switch a second circuit which can be completely separate from the first. For example a low voltage battery circuit can use a relay to switch a 230V AC mains circuit. There is no electrical connection inside the relay between the two circuits; the link is magnetic and mechanical.
The coil of a relay passes a relatively large current, typically 30mA for a 12V relay, but it can be as much as 100mA for relays designed to operate from lower voltages. Most ICs (chips) cannot provide this current and a transistor is usually used to amplify the small IC current to the larger value required for the relay coil. The maximum output current for the popular 555 timer IC is 200mA so these devices can supply relay coils directly without amplification.
Relays are usually SPDT or DPDT but they can have many more sets of switch contacts, for example relays with 4 sets of changeover contacts are readily available. For further information about switch contacts and the terms used to describe them please see the page on switches.

Most relays are designed for PCB mounting but you can solder wires directly to the pins providing you take care to avoid melting the plastic case of the relay.
The supplier's catalogue should show you the relay's connections. The coil will be obvious and it may be connected either way round. Relay coils produce brief high voltage 'spikes' when they are switched off and this can destroy transistors and ICs in the circuit. To prevent damage you must connect a protection diode across the relay coil.
The figure shows a relay with its coil and switch contacts. You can see a lever on the left being attracted by magnetism when the coil is switched on. This lever moves the switch contacts.

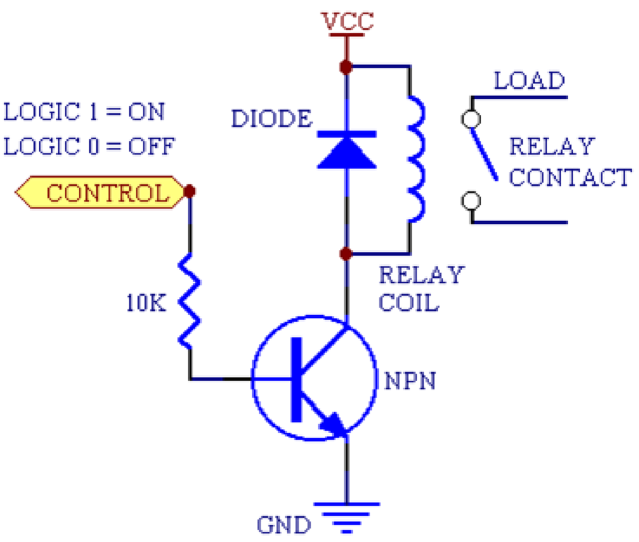


Figure 3.13: Circuit Diagram Of Relay

There is one set of contacts (SPDT) in the foreground and another behind them, making the relay DPDT.
The relay's switch connections are usually labelled COM, NC and NO:

- COM = Common, always connect to this; it is the moving part of the switch.
- NC = Normally Closed, COM is connected to this when the relay coil is off.
- NO = Normally Open, COM is connected to this when the relay coil is on.

3.8 Liquid Crystal Display (LCD)

This is the first interfacing example for the Parallel Port. We will start with something simple. This example doesn't use the Bi-directional feature found on newer ports, thus it should work with most, if no all Parallel Ports. It however doesn't show the use of the Status Port as an input. For a 16 Character x 2 Line LCD Module to the Parallel Port. These LCD Modules are very common these days, and are quite simple to work with, as all the logic required running them is on board.

3.8.1 Lcd Background:

Frequently, an 8051 program must interact with the outside world using input and output devices that communicate directly with a human being. One of the most common devices attached to an 8051 is an LCD display. Some of the most common LCDs connected to the 8051 are 16x2 and 20x2 displays. This means 16 characters per line by 2 lines and 20 characters per line by 2 lines, respectively.
Fortunately, a very popular standard exists which allows us to communicate with the vast majority of LCDs regardless of their manufacturer. The standard is referred to as HD44780U, which refers to the controller chip which receives data from an external source (in this case, the 8051) and communicates directly with the LCD.

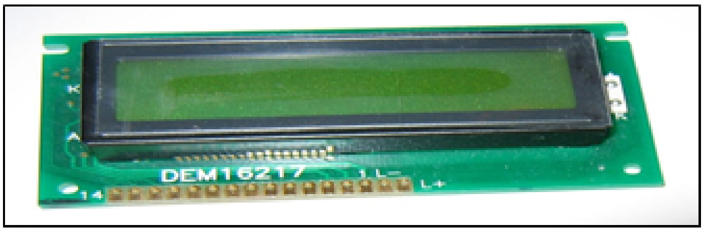


Figure 3.14: LCD

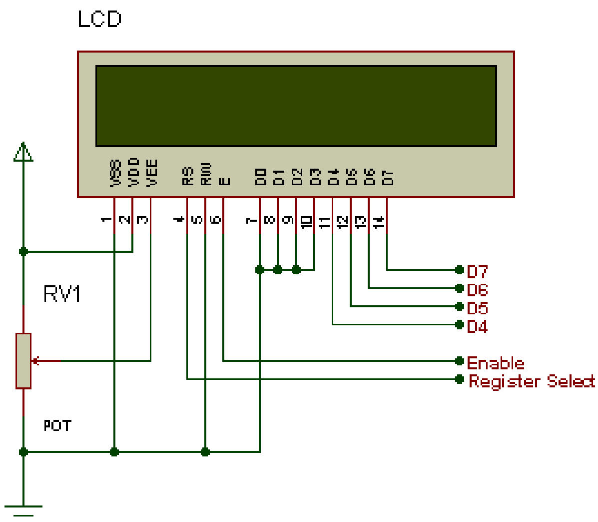


Figure 3.15: Block Diagram Of LCD

3.8.2 44780 LCD Background

The 44780 standard requires 3 control lines as well as either 4 or 8 I/O lines for the data bus. The user may select whether the LCD is to operate with a 4-bit data bus or an 8-bit data bus. If a 4-bit data bus is used the LCD will require a total of 7 data lines (3 control lines plus the 4 lines for the data bus). If an 8-bit data bus is used the LCD will require a total of 11 data lines (3 control lines plus the 8 lines for the data bus).
The three control lines are referred to as EN, RS, and RW. The EN line is called ”Enable.” This control line is used to tell the LCD that you are sending it data. To send data to the LCD, your program should make sure this line is low (0) and then set the other two control lines and/or put data on the data bus. When the other lines are completely ready, bring EN high (1) and wait for the minimum amount of time required by the LCD datasheet (this varies from LCD to LCD), and end by bringing it low (0) again.
The RS line is the ”Register Select” line. When RS is low (0), the data is to be treated as a command or special instruction (such as clear screen, position cursor, etc.). When RS is high (1), the data being sent is text data which should be displayed on the screen. For example, to display the letter ”T” on the screen you would set RS high.
The RW line is the ”Read/Write” control line. When RW is low (0), the information on the data bus is being written to the LCD. When RW is high (1), the program is effectively querying (or reading) the LCD. Only one instruction (”Get LCD status”) is a read command. All others are write commands–so RW will almost always be low.
Finally, the data bus consists of 4 or 8 lines (depending on the mode of operation selected by the user). In the case of an 8-bit data bus, the lines are referred to as DB0, DB1, DB2, DB3, DB4, DB5, DB6, and DB7.

3.9 Light Emitting Diodes (LED)

LEDs are semiconductor devices. Like transistors, and other diodes, LEDs are made out of silicon. What makes an LED give off light are the small amounts of chemical impurities that are added to the silicon, such as gallium, arsenide, indium, and nitride. When current passes through the LED, it emits photons as a byproduct. Normal light bulbs produce light by heating a metal filament until its white hot. Because LEDs produce photons directly and not via heat, they are far more efficient than incandescent bulbs. Not long ago LEDs were only bright enough to be used as indicators on dashboards or electronic equipment. But recent advances have made LEDs bright enough to rival traditional lighting technologies. Modern LEDs can replace incandescent bulbs in almost any application.
LEDs are based on the semiconductor diode. When the diode is forward biased (switched on), electrons are able to recombine with holes and energy is released in the form of light. This effect is called electroluminescence and the color of the light is determined by the energy gap of the semiconductor. The LED is usually small in area (less than 1 mm2) with integrated optical components to shape its radiation pattern and assist in reflection.
LEDs present many advantages over traditional light sources including lower energy consumption, longer lifetime, improved robustness, smaller size and faster switching. However, they are relatively expensive and require more precise current and heat management than traditional light sources.

Figure 3.16: LED

Applications of LEDs are diverse. They are used as low-energy and also for replacements for traditional light sources in well-established applications such as indicators and automotive lighting.
The compact size of LEDs has allowed new text and video displays and sensors to be developed, while their high switching rates are useful in communications technology. So here the role of LED is to indicate the status of the components like relays and power circuit etc

3.10 IN4007 Diodes

Diodes are used to convert AC into DC these are used as half wave rectifier or full wave rectifier. Three points must he kept in mind while using any type of diode.

1. Maximum forward current capacity
2. Maximum reverse voltage capacity
3. Maximum forward voltage capacity

Figure 3.17: DIODE

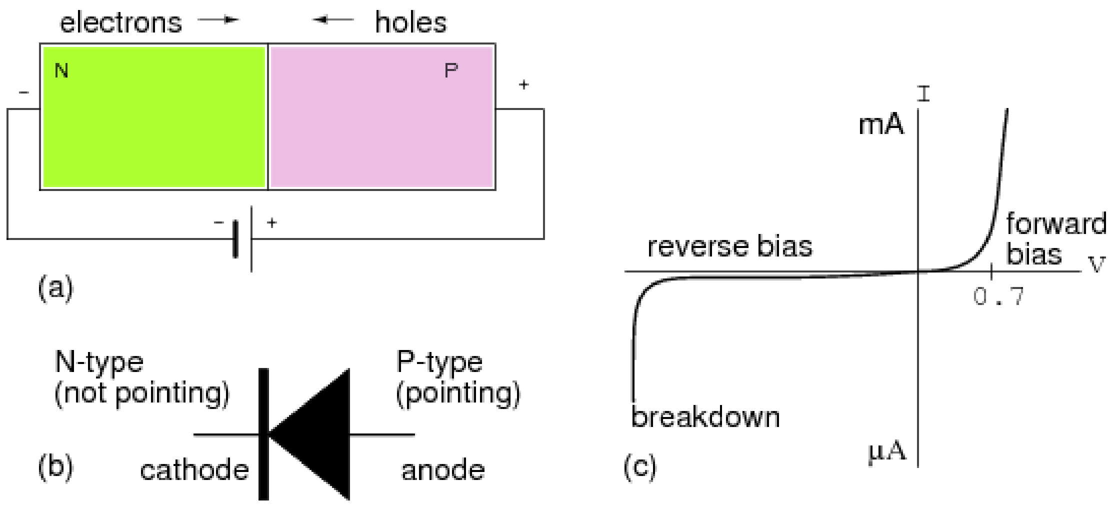


Figure 3.18: Block Diagram Of Diode

The number and voltage capacity of some of the important diodes available in the market are as follows:

- Diodes of number IN4001, IN4002, IN4003, IN4004, IN4005, IN4006 and IN4007 have maximum reverse bias voltage capacity of 50V and maximum forward current capacity of 1 Amp.
- Diode of same capacities can be used in place of one another. Besides this diode of more capacity can be used in place of diode of low capacity but diode of low capacity cannot

be used in place of diode of high capacity. For example, in place of IN4002; IN4001 or IN4007 can be used but IN4001 or IN4002 cannot be used in place of IN4007.The diode BY125made by company BEL is equivalent of diode from IN4001 to IN4003. BY 126 is equivalent to diodes IN4004 to 4006 and BY 127 is equivalent to diode IN4007.

3.10.1 PN Junction Operation

Now that you are familiar with P- and N-type materials, how these materials are joined together to form a diode, and the function of the diode, let us continue our discussion with the operation of the PN junction. But before we can understand how the PN junction works, we must first consider current flow in the materials that make up the junction and what happens initially within the junction when these two materials are joined together.

3.10.2 Current Flow In The N-Type Material

Conduction in the N-type semiconductor, or crystal, is similar to conduction in a copper wire. That is, with voltage applied across the material, electrons will move through the crystal just as current would flow in a copper wire. This is shown in figure 1-15. The positive potential of the battery will attract the free electrons in the crystal. These electrons will leave the crystal and flow into the positive terminal of the battery. As an electron leaves the crystal, an electron from the negative terminal of the battery will enter the crystal, thus completing the current path. Therefore, the majority current carriers in the N-type material (electrons) are repelled by the negative side of the battery and move through the crystal toward the positive side of the battery.

3.10.3 Current Flow In The P-Type Material

Current flow through the P-type material is illustrated. Conduction in the P material is by positive holes, instead of negative electrons. A hole moves from the positive terminal of the P material to the negative terminal. Electrons from the external circuit enter the negative terminal of the material and fill holes in the vicinity of this terminal. At the positive terminal, electrons are removed from the covalent bonds, thus creating new holes. This process continues as the steady stream of holes (hole current) moves toward the negative terminal.

3.11 Resistors

A resistor is a two-terminal electronic component designed to oppose an electric current by producing a voltage drop between its terminals in proportion to the current, that is, in accordance with Ohm's law:

$$V = IR$$

Figure 3.19: Resistor

Resistors are used as part of electrical networks and electronic circuits. They are extremely commonplace in most electronic equipment. Practical resistors can be made of various compounds and films, as well as resistance wire (wire made of a high-resistivity alloy, such as nickel/chrome).
The primary characteristics of resistors are their resistance and the power they can dissipate. Other characteristics include temperature coefficient, noise, and inductance. Less well-known is critical resistance, the value below which power dissipation limits the maximum permitted current flow, and above which the limit is applied voltage. Critical resistance depends upon the materials constituting the resistor as well as its physical dimensions; it's determined by design.
Resistors can be integrated into hybrid and printed circuits, as well as integrated circuits. Size, and position of leads (or terminals) are relevant to equipment designers; resistors must be physically large enough not to overheat when dissipating their power.
A resistor is a two-terminal passive electronic component which implements electrical resistance as a circuit element. When a voltage V is applied across the terminals of a resistor, a current I will flow through the resistor in direct proportion to that voltage. The reciprocal of the constant of proportionality is known as the resistance R, since, with a given voltage V, a larger value of R further "resists" the flow of current I as given by Ohm's law:

$$I = \frac{V}{R}$$

Figure 3.20: Ohm's Law

Resistors are common elements of electrical networks and electronic circuits and are ubiquitous in most electronic equipment. Practical resistors can be made of various compounds and films, as well as resistance wire (wire made of a high-resistivity alloy, such as nickel-chrome).

Resistors are also implemented within integrated circuits, particularly analog devices, and can also be integrated into hybrid and printed circuits.

3.11.1 Units:

The ohm (symbol:) is the SI unit of electrical resistance, named after Georg Simon Ohm. An ohm is equivalent to a volt per ampere. Since resistors are specified and manufactured over a very large range of values, the derived units of milliohm (1 m = 103), kilohm (1 k = 103), and megohm (1 M = 106) are also in common usage.
The reciprocal of resistance R is called conductance G = 1/R and is measured in Siemens (SI unit), sometimes referred to as a mho. Thus a Siemens is the reciprocal of an ohm: S = 1. Although the concept of conductance is often used in circuit analysis, practical resistors are always specified in terms of their resistance (ohms) rather than conductance.

3.11.2 Theory Of Operations

Ohm's law

The behavior of an ideal resistor is dictated by the relationship specified in Ohm's law:

$$V = I \cdot R$$

Figure 3.21: Ohm's Law

Ohm's law states that the voltage (V) across a resistor is proportional to the current (I) passing through it, where the constant of proportionality is the resistance (R).
Equivalently, Ohm's law can be stated:

$$I = \frac{V}{R}$$

Figure 3.22: Ohm's Law

This formulation of Ohm's law states that, when a voltage (V) is present across a resistance (R), a current (I) will flow through the resistance. This is directly used in practical computations. For example, if a 300 ohm resistor is attached across the terminals of a 12 volt battery, then a current of 12 / 300 = 0.04 amperes (or 40 milliamperes) will flow through that resistor.

Series and parallel resistors

In a series configuration, the current through all of the resistors is the same, but the voltage across each resistor will be in proportion to its resistance. The potential difference (voltage) seen across the network is the sum of those voltages, thus the total resistance can be found as the sum of those resistances:

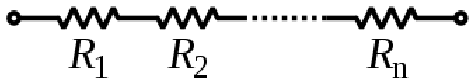
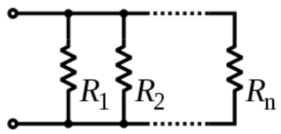


Figure 3.23: Equation of Series

As a special case, the resistance of N resistors connected in series, each of the same resistance R, is given by NR.
Resistors in a parallel configuration are each subject to the same potential difference (voltage), however the currents through them add. The conductances of the resistors then add to determine the conductance of the network. Thus the equivalent resistance (Req) of the network can be computed:



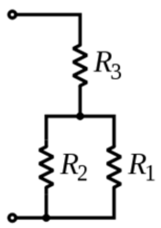
$$\frac{1}{R_{\mathrm{eq}}} = \frac{1}{R_1} + \frac{1}{R_2} + \cdots + \frac{1}{R_n}$$

$$R_{\mathrm{eq}} = R_1 \| R_2 = \frac{R_1 R_2}{R_1 + R_2}$$

Figure 3.24: Equation Parallel

The parallel equivalent resistance can be represented in equations by two vertical lines "——" (as in geometry) as a simplified notation. For the case of two resistors in parallel, this can be calculated using:

As a special case, the resistance of N resistors connected in parallel, each of the same resistance R, is given by R/N.
A resistor network that is a combination of parallel and series connections can be broken up into smaller parts that are either one or the other. For instance,



$$R_{\mathrm{eq}} = (R_1 \| R_2) + R_3 = \frac{R_1 R_2}{R_1 + R_2} + R$$

Figure 3.25: Equation For Series And Parallel

3.12 Capacitors

A capacitor or condenser is a passive electronic component consisting of a pair of conductors separated by a dielectric. When a voltage potential difference exists between the conductors, an electric field is present in the dielectric. This field stores energy and produces a mechanical force between the plates. The effect is greatest between wide, flat, parallel, narrowly separated conductors.

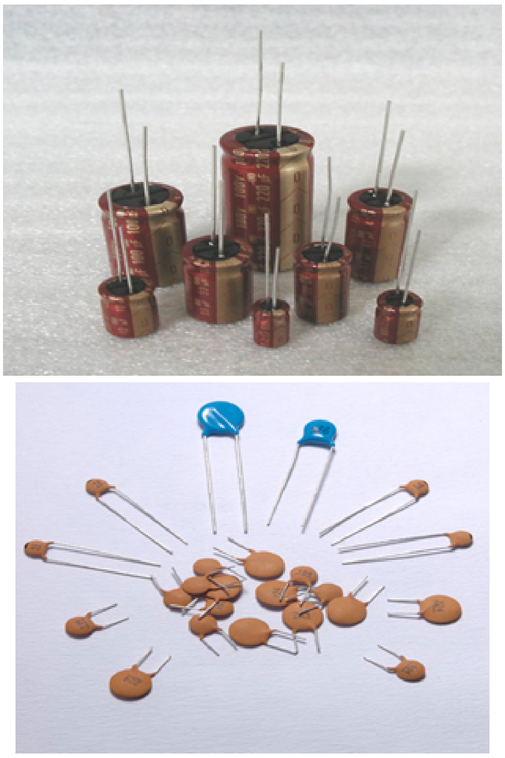


Figure 3.26: Capacitor

An ideal capacitor is characterized by a single constant value, capacitance, which is measured in farads. This is the ratio of the electric charge on each conductor to the potential difference between them. In practice, the dielectric between the plates passes a small amount of leakage current. The conductors and leads introduce an equivalent series resistance and the dielectric has an electric field strength limit resulting in a breakdown voltage.
The properties of capacitors in a circuit may determine the resonant frequency and quality factor of a resonant circuit, power dissipation and operating frequency in a digital logic circuit, energy capacity in a high-power system, and many other important aspects.
A capacitor (formerly known as condenser) is a device for storing electric charge. The forms of practical capacitors vary widely, but all contain at least two conductors separated by a non-conductor. Capacitors used as parts of electrical systems, for example, consist of metal foils separated by a layer of insulating film.
Capacitors are widely used in electronic circuits for blocking direct current while allowing alternating current to pass, in filter networks, for smoothing the output of power supplies, in the resonant circuits that tune radios to particular frequencies and for many other purposes.
A capacitor is a passive electronic component consisting of a pair of conductors separated by a dielectric (insulator). When there is a potential difference (voltage) across the conductors, a static electric field develops in the dielectric that stores energy and produces a mechanical

force between the conductors. An ideal capacitor is characterized by a single constant value, capacitance, measured in farads. This is the ratio of the electric charge on each conductor to the potential difference between them.
The capacitance is greatest when there is a narrow separation between large areas of conductor, hence capacitor conductors are often called "plates", referring to an early means of construction. In practice the dielectric between the plates passes a small amount of leakage current and also has an electric field strength limit, resulting in a breakdown voltage, while the conductors and leads introduce an undesired inductance and resistance.

3.12.1 Theory Of Operations

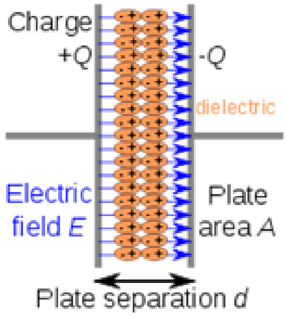


Figure 3.27: Capacitance

Charge separation in a parallel-plate capacitor causes an internal electric field. A dielectric (orange) reduces the field and increases the capacitance.

Figure 3.28: Parallel Capacitor

A simple demonstration of a parallel-plate capacitor
A capacitor consists of two conductors separated by a non-conductive region. The non-conductive region is called the dielectric or sometimes the dielectric medium. In simpler terms, the dielectric is just an electrical insulator. Examples of dielectric mediums are glass, air, paper, vacuum, and even a semiconductor depletion region chemically identical to the conductors. A capacitor is assumed to be self-contained and isolated, with no net electric charge and no influence from any external electric field. The conductors thus hold equal and opposite charges on their facing surfaces, and the dielectric develops an electric field. In SI units, a

capacitance of one farad means that one coulomb of charge on each conductor causes a voltage of one volt across the device. The capacitor is a reasonably general model for electric fields within electric circuits. An ideal capacitor is wholly characterized by a constant capacitance C, defined as the ratio of charge Q on each conductor to the voltage V between them:

$$C = \frac{Q}{V}$$

Figure 3.29: Equation for Capacitance

Sometimes charge build-up affects the capacitor mechanically, causing its capacitance to vary. In this case, capacitance is defined in terms of incremental changes:

$$C = \frac{\mathrm{d}q}{\mathrm{d}v}$$

Figure 3.30: Capacitance in terms of incremental changes

3.12.2 Parallel Plate Mode

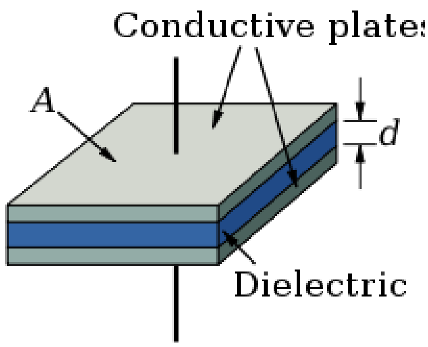


Figure 3.31: Parallel plate mode

Dielectric is placed between two conducting plates, each of area A and with a separation of d.
The simplest capacitor consists of two parallel conductive plates separated by a dielectric with permittivity (such as air). The model may also be used to make qualitative predictions for other device geometries. The plates are considered to extend uniformly over an area A and a charge density = Q/A exists on their surface. Assuming that the width of the plates is much greater than their separation d, the electric field near the centre of the device will be uniform with the magnitude E = /. The voltage is defined as the line integral of the electric field between the plates

Solving this for C = Q/V reveals that capacitance increases with area and decreases with separation

$$V = \int_0^d E\mathrm{d}z = \int_0^d \frac{\rho}{\varepsilon}\mathrm{d}z = \frac{\rho d}{\varepsilon} = \frac{Qd}{\varepsilon A}.$$

Figure 3.32: Line integral

$$C = \frac{\varepsilon A}{d}$$

Figure 3.33: Capacitance increases

The capacitance is therefore greatest in devices made from materials with a high permittivity.

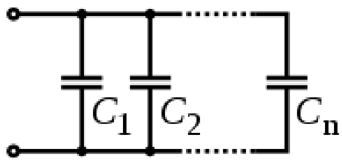


Figure 3.34: Capacitor in parallel

Chapter 4

Load Cell

Day-in and day-out we have been using electronic scales or weighing machines. You go to any grocery store or a jewellery shop; the items are weighed using electronic weighing machines (Weighing balances are also used in situations where very high accuracy is not of paramount importance). But, have we ever given a thought on the mechanism of the electronic scales?
At the heart of electronic scales or weighing machines is a sensor called load cell. These sensors sense the force (or weight) of the items and the electronic circuitry processes the sensors output and displays it on the indicator. Load cells are highly accurate transducers which provide the user with information not generally obtainable by other technology due to commercial factors.

Figure 4.1: Image of load cell

Usage of load cell is not limited to electronics scales; they are used load testing machines, industrial scales, flow-meters, etc., though we hardly ever come in direct contact with the load cells. In short, load cell can be used wherever there is a requirement of force measurement.
Subsequent sections will discuss in-outs of the load cells.

4.1 What Is Load Cell

As per dictionary, a load cell is described as a weight measurement device necessary for electronic scales that display weights in digits. However, load cell is not restricted to weight measurement in electronic scales. Load cell is a passive transducer or sensor which converts applied force into electrical signals. They are also referred to as Load transducers.

Load cells use different operating principles, viz.,

- Load Cells based on fluid pressure
- Load Cells based on elasticity
- Load Cells based on magnetostriction effect or piezoelectric effect

However, the only load cells which are prevalent are the load cells based on strain gages. Hence, the term load cell means strain gage-based load cells. The reason behind the wide adoption of strain gage-based load cells is their characteristics

1. Highly precise and linear measurements.
2. Little influence due to temperature changes.
3. Small size compared with other types of load cells.
4. Long operating life due to lack of moving parts or any parts that generate friction.
5. Ease in production due to small number of components.
6. Excellent fatigue characteristics .

Chapter 5

Arduino

Arduino is an open-source prototyping platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger on a button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based onWiring), and the Arduino Software (IDE), based on Processing.
Over the years Arduino has been the brain of thousands of projects, from everyday objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals - has gathered around this open-source platform, their contributions have added up to an incredible amount of accessible knowledge that can be of great help to novices and experts alike.
Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and programming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them independently and eventually adapt them to their particular needs. The software, too, is open-source, and it is growing through the contributions of users worldwide.

5.1 Why Arduino?

Thanks to its simple and accessible user experience, Arduino has been used in thousands of different projects and applications. The Arduino software is easy-to-use for beginners, yet flexible enough for advanced users. It runs on Mac, Windows, and Linux. Teachers and students use it to build low cost scientific instruments, to prove chemistry and physics principles, or to get started with programming and robotics. Designers and architects build interactive prototypes, musicians and artists use it for installations and to experiment with new musical instruments. Makers, of course, use it to build many of the projects exhibited at the Maker Faire, for example. Arduino is a key tool to learn new things. Anyone - children, hobbyists, artists, programmers - can start tinkering just following the step by step instructions of a kit, or sharing ideas online with other members of the Arduino community.
There are many other microcontrollers and microcontroller platforms available for physical computing. Parallax Basic Stamp, Netmedia's BX-24, Phidgets, MIT's Handyboard, and many others offer similar functionality. All of these tools take the messy details of microcon-

troller programming and wrap it up in an easy-to-use package. Arduino also simplifies the process of working with microcontrollers, but it offers some advantage for teachers, students, and interested amateurs over other systems:

- **Inexpensive** - Arduino boards are relatively inexpensive compared to other microcontroller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than 3350rs.
- **Cross-platform** - The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most microcontroller systems are limited to Windows.
- **Simple, clear programming environment** - The Arduino Software (IDE) is easy-to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.
- **Open source and extensible software** - The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from Arduino to the AVR C programming language on which it's based. Similarly, you can add AVR-C code directly into your Arduino programs if you want to.
- **Open source and extensible hardware** - The plans of the Arduino boards are published under a Creative Commons license, so experienced circuit designers can make their own version of the module, extending it and improving it. Even relatively inexperienced users can build the breadboard version of the module in order to understand how it works and save money.

Arduino Uno

The most common version of Arduino is the Arduino Uno. This board is what most people are talking about when they refer to an Arduino. In the next step, there is a more complete rundown of its features.

Arduino NG Diecimil aand the Duemilanove Legacy Versions

Legacy versions of the Arduino Uno product line consist of the NG, Diecimila, and the Duemilanove. The important thing to note about legacy boards is that they lack particular feature of the Arduino Uno. Some key differences:

- The Diecimila and NG use an ATMEGA168 chips (as opposed to the more powerful ATMEGA328),
- Both the Diecimila and NG have a jumper next to the USB port and require manual selection of either USB or battery power.
- The Arduino NG requires that you hold the rest button on the board for a few seconds prior to uploading a program.

Chapter 6

Brief Description Of Working Of Relay

A relay is an electrically operated switch. Current flowing through the coil of the relay creates a magnetic field which attracts a lever and changes the switch contacts. The coil current can be on or off so relays have two switch positions and most have double throw (changeover) switch contacts. Relays allow one circuit to switch a second circuit which can be completely separate from the first. For example a low voltage battery circuit can use a relay to switch a 230V AC mains circuit. There is no electrical connection inside the relay between the two circuits; the link is magnetic and mechanical.

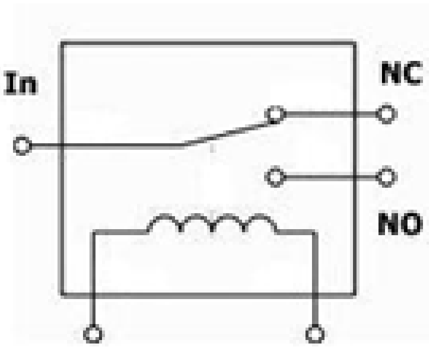


Figure 6.1: Block diagram of relay

6.1 ULN 2003 Relay Driver IC

ULN2003 is an IC which is used to interface relay with the microcontroller since the output of the micro controller is maximum 5V with too little current delivery and is not practicable to operate a relay with that voltage. ULN2003 is a relay driver IC consisting of a set of Darlington transistors. If logic high is given to the IC as input then its output will be logic low but not the vice versa. Here in ULN2003 pin 1 to 7 are IC inputs and 10 to 16 are IC outputs. If logic 1 is given to its pin no 1 the corresponding pin 16 goes low. If a relay coil is connected from +ve to the output pin of the uln2003,(the relay driver) then the relay contacts change their position from normally open to close the circuit as shown below for the load on (say a lamp to start glowing). If logic 0 is given at the input the relay switches off. Similarly upto seven relays can be used for seven different loads to be switched on by the normally open(NO) contact or switched off by the normally closed contact(NC).

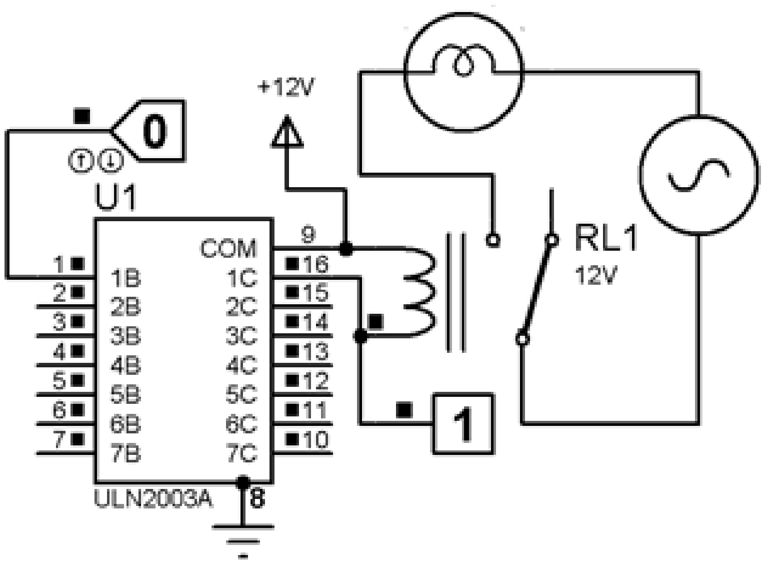


Figure 6.2: Load Off

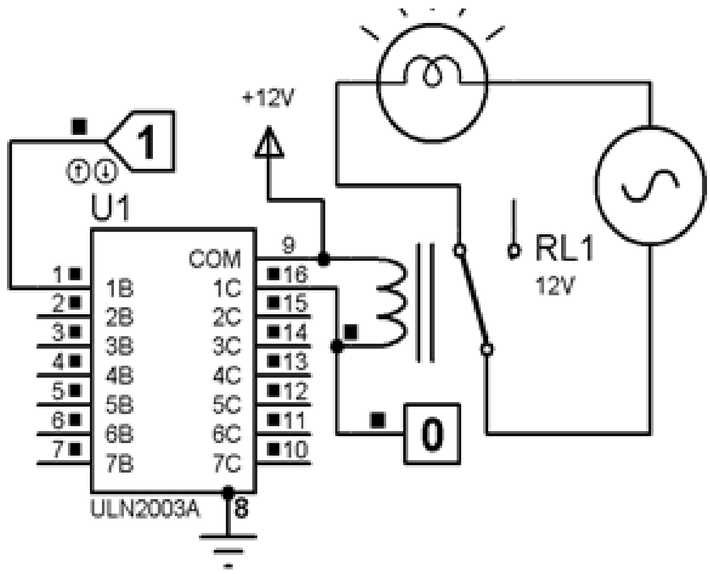


Figure 6.3: Load On

Chapter 7

Circuit Diagram

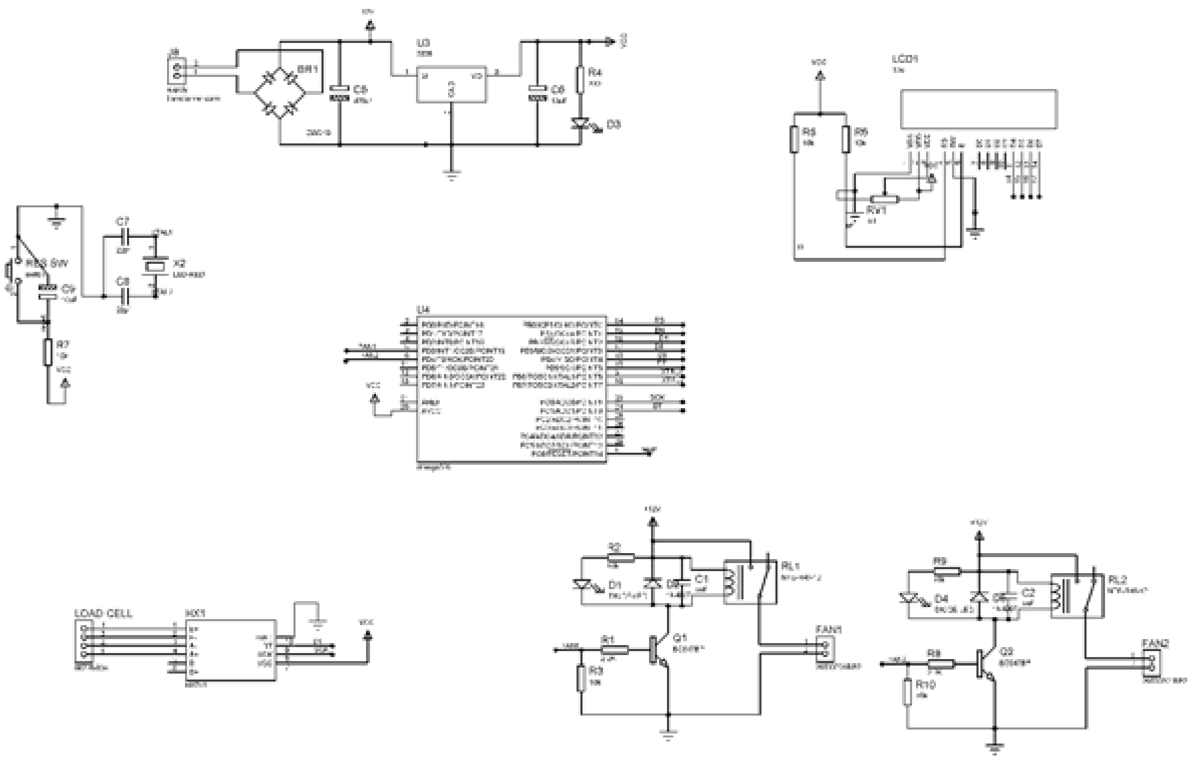


Figure 7.1: Layout Diagram

Chapter 8

Bill Of Material

Category	Quantity	References	Value
Capacitors	1	C1	470f
Capacitors	5	C2,C6-C9	104
Capacitors	2	C3-C4	22p
Capacitors	1	C5	10uf
Resistors	1	R1	330
Resistors	5	R2-R4,R9-R10	10k
Resistors	1	RN1	10k resistor network
Resistors	1	RV1	10k
Resistors	4	R5-R8	1k
Integrated Circuits	1	U1	ATMEGA328P
Integrated Circuits	1	U2	7805
Integrated Circuits	2	HX1-HX2	HX711 MODULE
Integrated Circuits	1	U3	ULN2003A
Diodes	1	D1	red
Diodes	4	D2,D4-D5,D8	1N4007
Diodes	4	D3,D6-D7,D9	DIODE-LED
Miscellaneous	2	12V	2 pin male connector
Miscellaneous	1	BR1	2W04G
Miscellaneous	1	FUSE1	FUSE
Miscellaneous	1	LCD1	16*2
Miscellaneous	1	LCD1	16 PIN FEMALE CONNECTOR
Miscellaneous	5	LOAD1-LOAD4,j1	2 pin PBT
Miscellaneous	2	LOAD CELL 1-LOAD CELL 2	2 pin male connector
Miscellaneous	2	LOAD CELL 1-LOAD CELL 2	load cells
Miscellaneous	1	RES SW	push button
Miscellaneous	4	RL1-RL4	12v relay
Miscellaneous	2	SW1-SW2	SPDT slide switch
Miscellaneous	1	X1	16MHZ CRYSTAL

Figure 8.1: Bill of material

Category	Quantity
Capacitors	9
Resistors	12
Integrated Circuits	5
Diodes	9
Miscellaneous	23
Total	**58**

Figure 8.2: Total number of material

Chapter 9

Coding

9.1 Arduino Ide

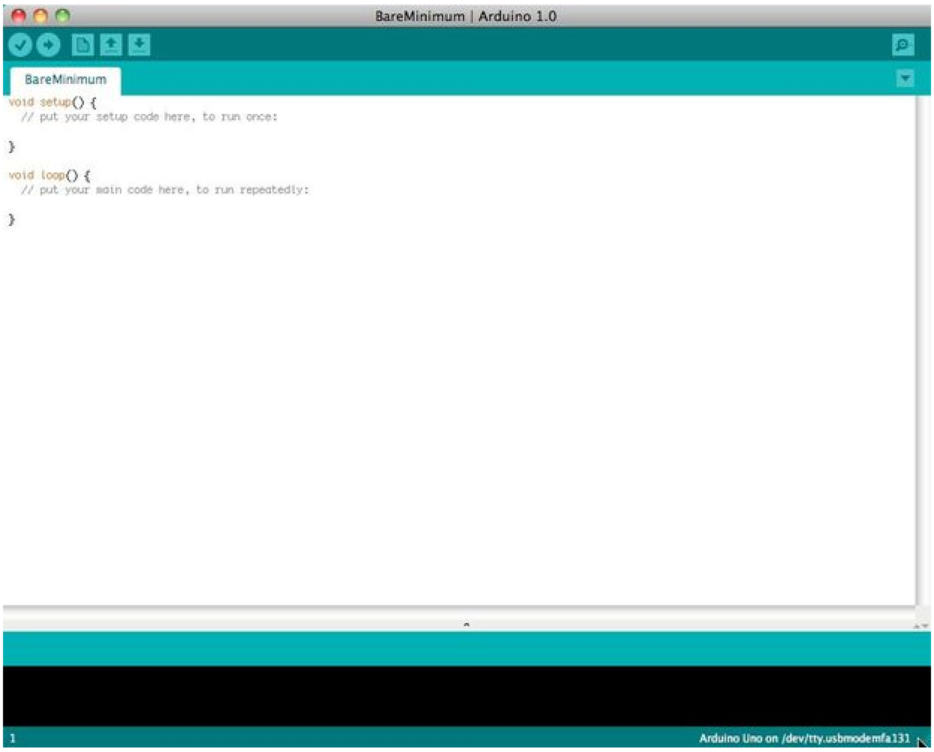


Figure 9.1: Arduino Gui

Before you can start doing anything with the Arduino, you need to download and install the Arduino IDE (integrated development environment). From this point on we will be referring to the Arduino IDE as the Arduino Programmer.

The Arduino Programmer is based on the Processing IDE and uses a variation of the C and C++ programming languages.

9.2 Setting

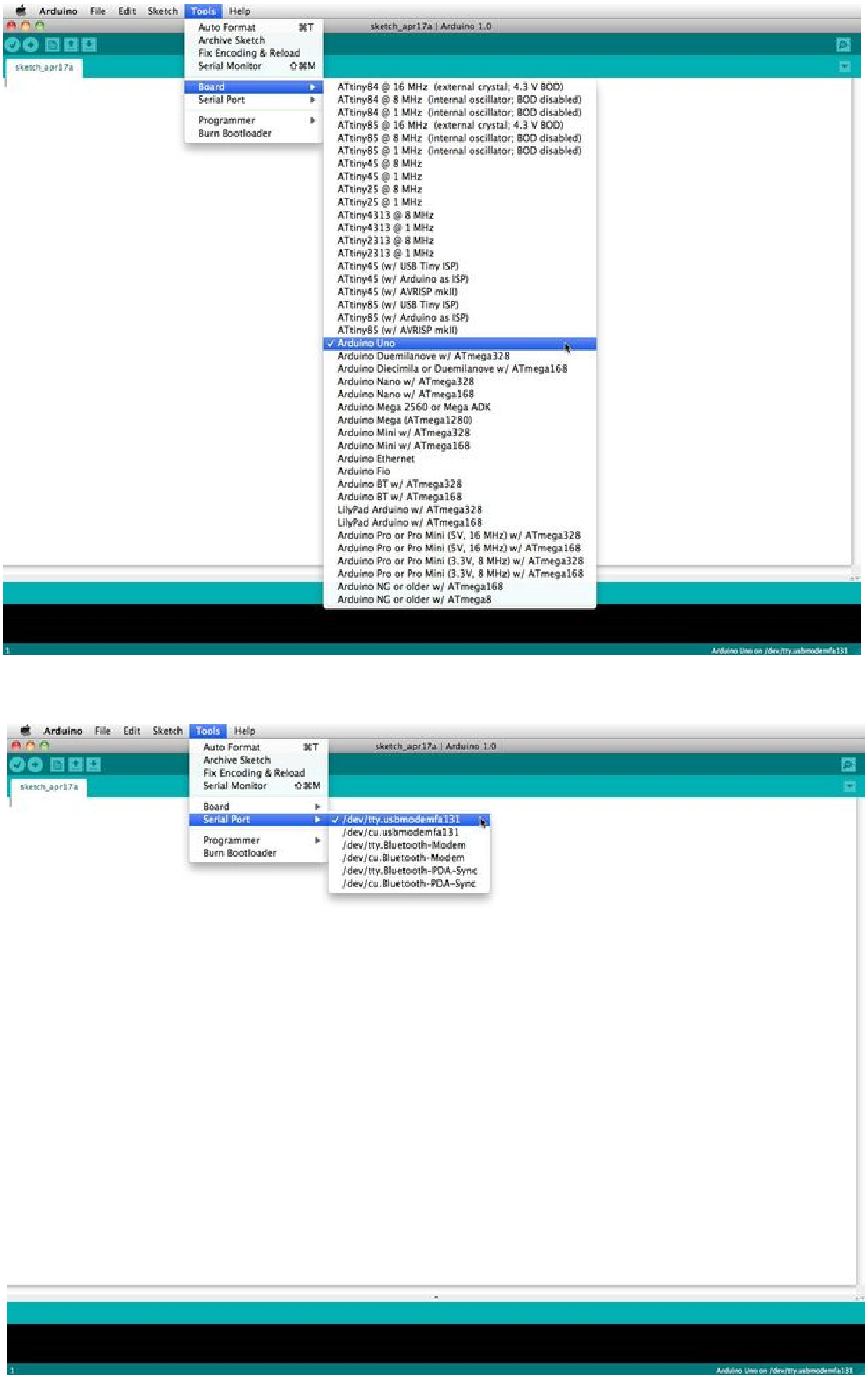


Figure 9.2: Tolls

Before you can start doing anything in the Arduino programmer, you must set the board-type and serial port.

To set the board, go to the following:

Tools→ Boards
Select the version of board that you are using. Since I have an Arduino Uno plugged in, I obviously selected "Arduino Uno."

To set the serial port, go to the following:

Tools→ Serial Port
Select the serial port that looks like:
/dev/tty.usbmodem [random numbers]

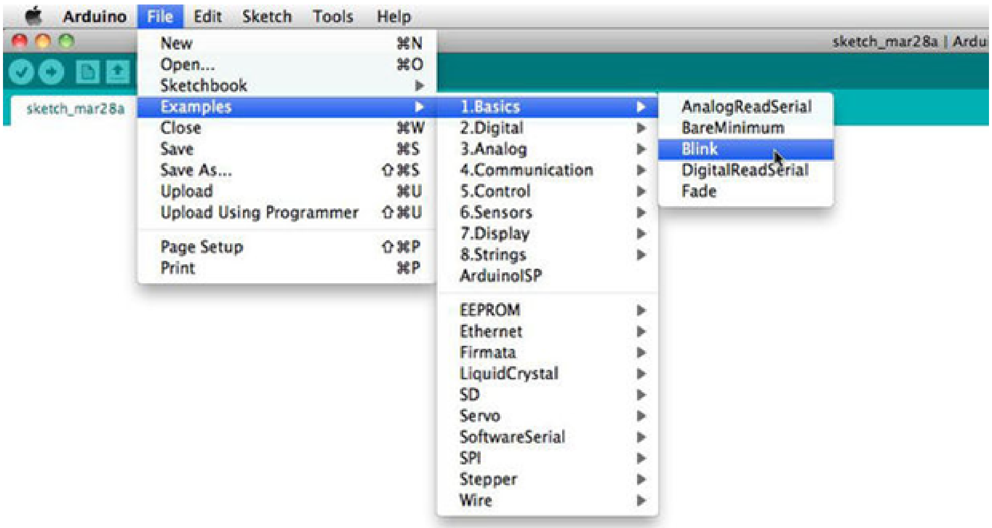


Figure 9.3: Arduino Uno

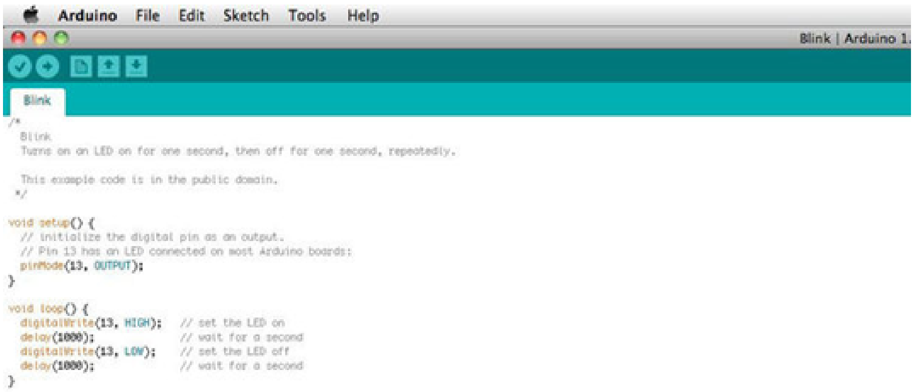


Figure 9.4: Blink

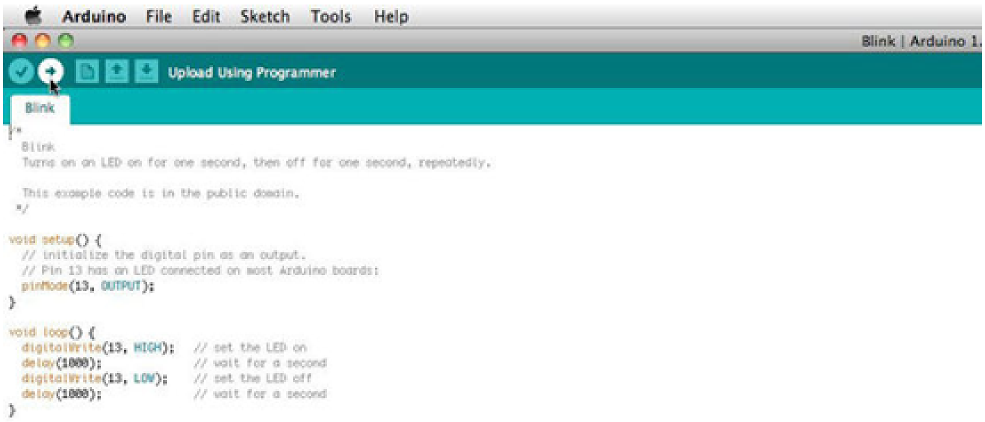


Figure 9.5: blink 1

Arduino programs are called sketches. The Arduino programmer comes with a ton of example sketches preloaded. This is great because even if you have never programmed anything in your life, you can load one of these sketches and get the Arduino to do something.
To get the LED tied to digital pin 13 to blink on and off, let's load the blink example.

The blink example can be found here:

Files→ Examples → Basics → Blink

The blink example basically sets pin D13 as an output and then blinks the test LED on the Arduino board on and off every second. Once the blink example is open, it can be installed onto the ATMEGA328 chip by pressing the upload button, which looks like an arrow pointing to the right. Notice that the surface mount status LED connected to pin 13 on the Arduino will start to blink. You can change the rate of the blinking by changing the length of the delay and pressing the upload button again.

Serial monitor

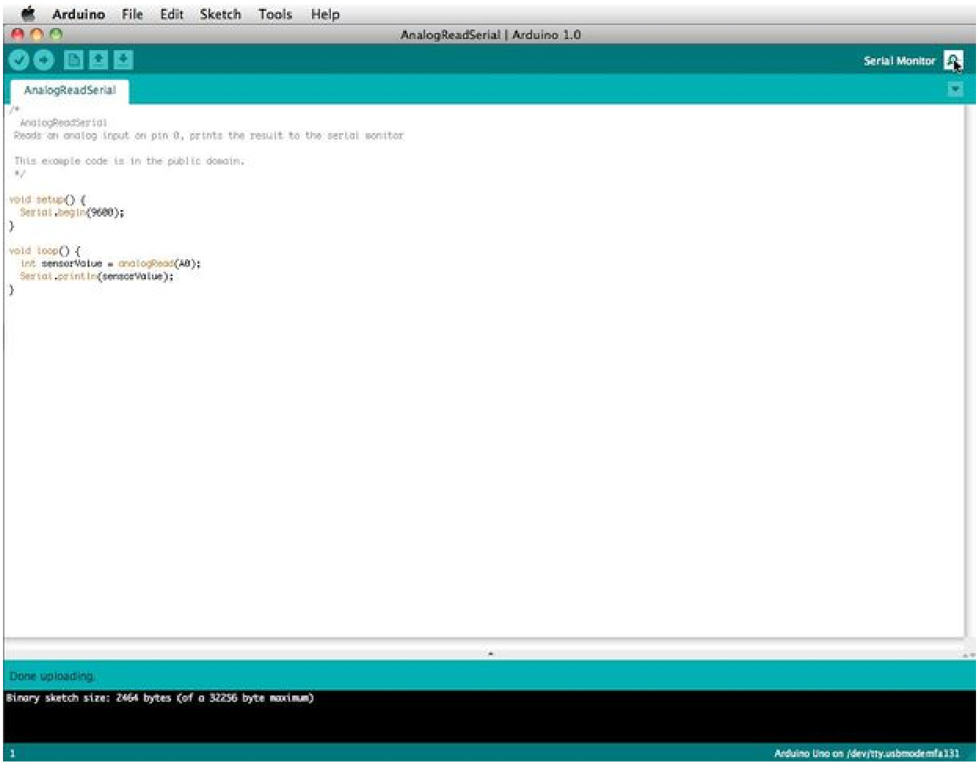


Figure 9.6: Analog Read Serial

The serial monitor allows your computer to connect serially with the Arduino. This is important because it takes data that your Arduino is receiving from sensors and other devices and displays it in real-time on your computer. Having this ability is invaluable to debug your code and understand what number values the chip is actually receiving.
For instance, connect center sweep (middle pin) of a potentiometer to A0, and the outer pins, respectively, to 5v and ground. Next upload the sketch shown below:

File→ Examples → 1.Basics → AnalogReadSerial

Click the button to engage the serial monitor which looks like a magnifying glass. You can now see the numbers being read by the analog pin in the serial monitor. When you turn the knob the numbers will increase and decrease.
The numbers will be between the range of 0 and 1023. The reason for this is that the analog

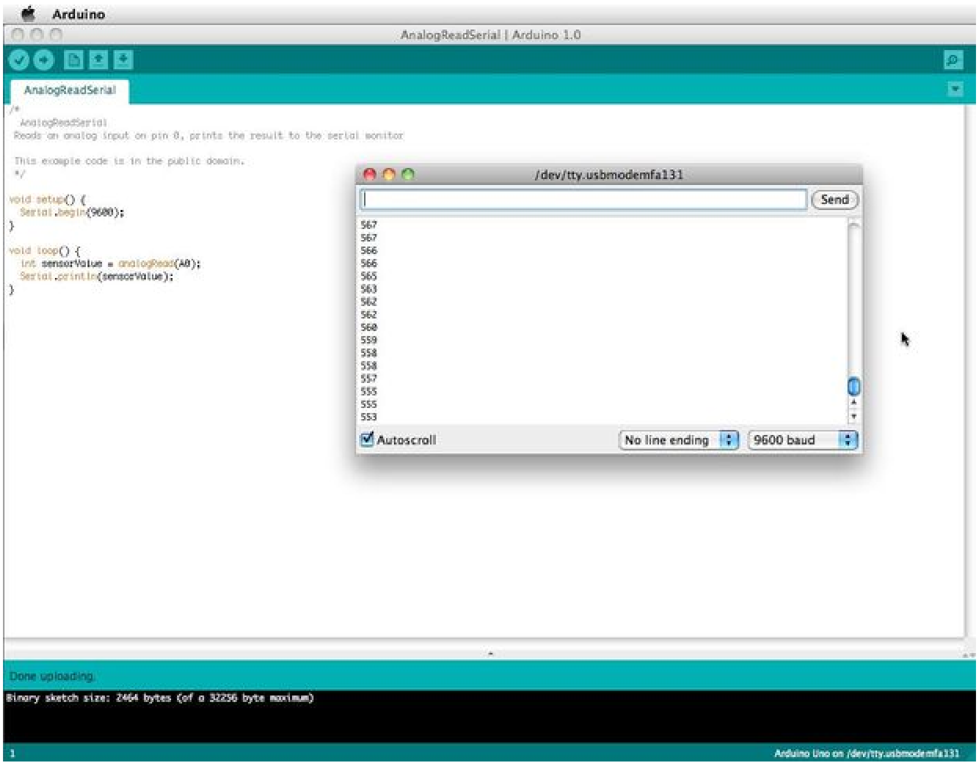


Figure 9.7: Analog Read Serial.

pin is converting a voltage between 0 and 5V to a discreet number.

The Arduino has two different types of input pins, those being analog and digital.

To begin with, lets look at the digital input pins.

Digital input pins only have two possible states, which are on or off. These two on and off states are also referred to as:

- HIGH or LOW
- 1 or 0
- 5V or 0V.

This input is commonly used to sense the presence of voltage when a switch is opened or closed. Digital inputs can also be used as the basis for countless digital communication protocols. By creating a 5V (HIGH) pulse or 0V (LOW) pulse, you can create a binary signal, the basis of all computing. This is useful for talking to digital sensors like a PING ultrasonic sensor, or communicating with other devices.
For a simple example of a digital input in use, connect a switch from digital pin 2 to 5V, a 10K resistor** from digital pin 2 to ground, and run the following code:

File→ Examples → 2.Digital → Button

Analog in

Aside from the digital input pins, the Arduino also boasts a number of analog input pins. Analog input pins take an analog signal and perform a 10-bit analog-to-digital (ADC) conversion to turn it into a number between 0 and 1023 (4.9mV steps).
This type of input is good for reading resistive sensors. These are basically sensors which provide resistance to the circuit. They are also good for reading a varying voltage signal between 0 and 5V. This is useful when interfacing with various types of analog circuitry.

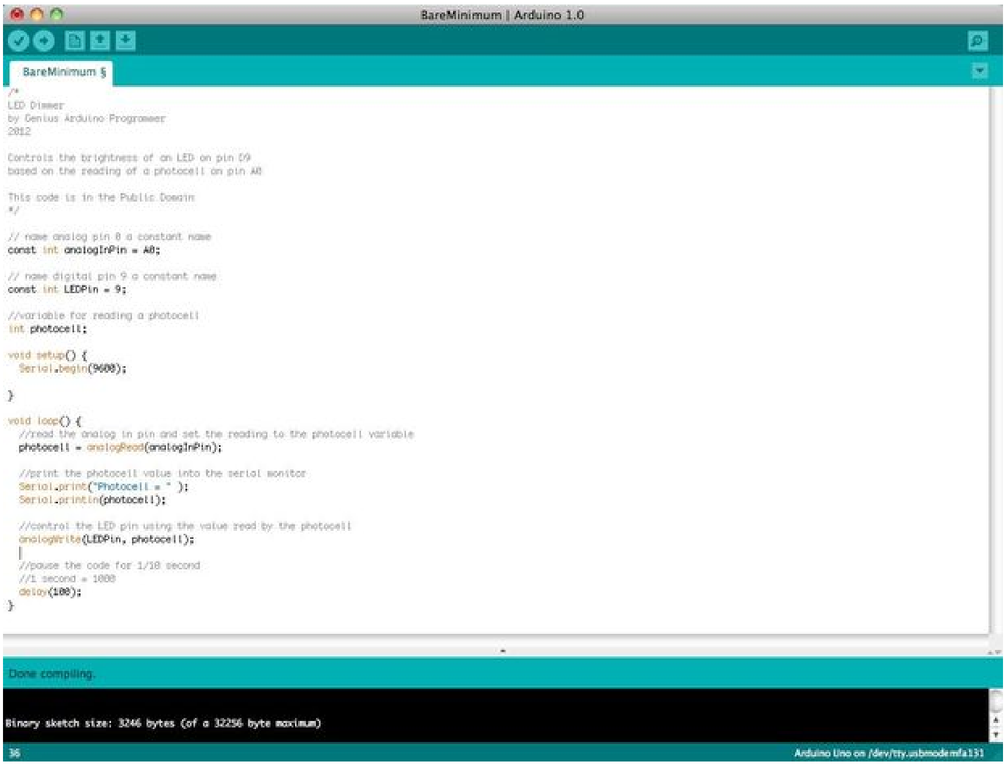


Figure 9.8: Analog In

To write your own code, you will need to learn some basic programming language syntax. In other words, you have to learn how to properly form the code for the programmer to understand it. You can think of this kind of like understanding grammar and punctuation. You can write an entire book without proper grammar and punctuation, but no one will be abler to understand it, even if it is in English.
Some important things to keep in mind when writing your own code:

- An Arduino program is called a sketch.
- All code in an Arduino sketch is processed from top to bottom.
- Arduino sketches are typically broken into five parts.

  1. The sketch usually starts with a header that explains what the sketch is doing, and who wrote it.
  2. Next, it usually defines global variables. Often, this is where constant names are given to the different Arduino pins.
  3. After the initial variables are set, the Arduino begins the setup routine. In the setup function, we set initial conditions of variables when necessary, and run any preliminary

code that we only want to run once. This is where serial communication is initiated, which is required for running the serial monitor.
4. From the setup function, we go to the loop routine. This is the main routine of the sketch. This is not only where your main code goes, but it will be executed over and over, so long as the sketch continues to run.
5. Below the loop routine, there is often other functions listed. These functions are user-defined and only activated when called in the setup and loop routine. When these functions are called, the Arduino processes all of the code in the function from top to bottom and then goes back to the next line in the sketch where it left off when the function was called. Functions are good because they allow you to run standard routines - over and over - without having to write the same lines of code over and over. You can simply call upon a function multiple times, and this will free up memory on the chip because the function routine is only written once. It also makes code easier to read. To learn how to form your own functions, check outthis page.

- All of that said, the only two parts of the sketch which are mandatory are the Setup and Loop routines.
- Code must be written in the Arduino Language, which is roughly based on C.
- Almost all statements written in the Arduino language must end with a ;
- Conditionals (such as if statements and for loops) do not need a ;
- Conditionals have their own rules and can be found under "Control Structures" on the Arduino Language page
- Variables are storage compartments for numbers. You can pass values into and out of variables. Variables must be defined (stated in the code) before they can be used and need to have a data type associated with it. To learn some of the basic data types, review the Language Page.

Okay! So let us say we want to write code that reads a photocell connected to pin A0, and use the reading we get from the photocell to control the brightness of an LED connected to pin D9.
First, we want to open the BareMinimum sketch, which can be found at:
File→ Examples → 1.Basic → BareMinimum

9.2.1 The BareMinimum Sketch should look like this:

```
<pre>void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
 Next, lets put a header on the code, so other people
```

```
 know about what we are making, why, and under what terms:

 <pre>/*
LED Dimmer
by Genius Arduino Programmer
2012

Controls the brightness of an LED on pin D9
based on the reading of a photocell on pin A0

This code is in the Public Domain
*/

void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}

Once that is all squared away, let us define the pin names,
and establish variables:\\

<pre>/*
LED Dimmer
by Genius Arduino Programmer
2012

Controls the brightness of an LED on pin D9
based on the reading of a photocell on pin A0

This code is in the Public Domain
*/

// name analog pin 0 a constant name
const int analogInPin = A0;

// name digital pin 9 a constant name
const int LEDPin = 9;

//variable for reading a photocell
int photocell;

void setup() {
  // put your setup code here, to run once:
```

```
}

void loop() {
  // put your main code here, to run repeatedly:

}
```

Now that variables and pin names are set, let us write the actual code:

```
<pre>/*
LED Dimmer
by Genius Arduino Programmer
2012

Controls the brightness of an LED on pin D9
based on the reading of a photocell on pin A0

This code is in the Public Domain
*/

// name analog pin 0 a constant name
const int analogInPin = A0;

// name digital pin 9 a constant name
const int LEDPin = 9;

//variable for reading a photocell
int photocell;

void setup() {
//nothing here right now

}

void loop() {
  //read the analog in pin and set the reading to the
  photocell variable
  photocell = analogRead(analogInPin);

  //control the LED pin using the value read by the photocell
  analogWrite(LEDPin, photocell);

  //pause the code for 1/10 second
  //1 second = 1000
  delay(100);
}
```

If we want to see what numbers the analog pin is actually reading from the photocell, we will need to use the serial monitor.

```
Let's activate the serial port and output those numbers:
\\
<pre>/*
LED Dimmer
by Genius Arduino Programmer
2012

Controls the brightness of an LED on pin D9
based on the reading of a photocell on pin A0

This code is in the Public Domain
*/

// name analog pin 0 a constant name
const int analogInPin = A0;

// name digital pin 9 a constant name
const int LEDPin = 9;

//variable for reading a photocell
int photocell;

void setup() {
  Serial.begin(9600);

}

void loop() {
  //read the analog in pin and set the reading to the photocell variable
  photocell = analogRead(analogInPin);

  //print the photocell value into the serial monitor
  Serial.print("Photocell = " );
  Serial.println(photocell);

  //control the LED pin using the value read by the photocell
  analogWrite(LEDPin, photocell);

  //pause the code for 1/10 second
  //1 second = 1000
  delay(100);
}
```

Chapter 10

Hardware Testing

10.1 Continuity Test:

In electronics, a continuity test is the checking of an electric circuit to see if current flows (that it is in fact a complete circuit). A continuity test is performed by placing a small voltage (wired in series with an LED or noise-producing component such as a piezoelectric speaker) across the chosen path. If electron flow is inhibited by broken conductors, damaged components, or excessive resistance, the circuit is "open".
Devices that can be used to perform continuity tests include multi meters which measure current and specialized continuity testers which are cheaper, more basic devices, generally with a simple light bulb that lights up when current flows. An important application is the continuity test of a bundle of wires so as to find the two ends belonging to a particular one of these wires; there will be a negligible resistance between the "right" ends, and only between the "right" ends.
This test is the performed just after the hardware soldering and configuration has been completed. This test aims at finding any electrical open paths in the circuit after the soldering. Many a times, the electrical continuity in the circuit is lost due to improper soldering, wrong and rough handling of the PCB, improper usage of the soldering iron, component failures and presence of bugs in the circuit diagram. We use a multi meter to perform this test. We keep the multi meter in buzzer mode and connect the ground terminal of the multi meter to the ground. We connect both the terminals across the path that needs to be checked. If there is continuation then you will hear the beep sound.

10.2 Power On Test:

This test is performed to check whether the voltage at different terminals is according to the requirement or not. We take a multi meter and put it in voltage mode. Remember that this test is performed without microcontroller. Firstly, we check the output of the transformer, whether we get the required 12 v AC voltage.
Then we apply this voltage to the power supply circuit. Note that we do this test without microcontroller because if there is any excessive voltage, this may lead to damaging the controller. We check for the input to the voltage regulator i.e., are we getting an input of 12v and an output of 5v. This 5v output is given to the microcontrollers 40th pin. Hence we check for the voltage level at 40th pin. Similarly, we check for the other terminals for the required voltage. In this way we can assure that the voltage at all the terminals is as per the requirement.

Chapter 11

Analaysis

11.1 Advantges:-

- It is very efficient.
- It prevents loss of electricity. *It does not depend on any other system or human being*

11.2 Disadvantages:-

- Costly: Cost is Rs 5000/- (approx.) for a single system. If an individual consumes it then the system will be expensive.
- A lot of wires required: This system requires a lot of wires which makes it difficult to maintain, although the wires will be packed.

11.3 Application :-

- It is very useful in offices.
- Also in the classrooms of schools and colleges.
- This system can be implemented for providing ease and comfort to the disabled/handicapped people.
- An individual can set this system in his/her own company.

11.4 Future Scope :-

- This system can be modified, in case if we come to know that our system is consuming more electricity, by attaching a solar panel to the system which will operate through solar power instead of electricity.
- Since wires are the problem in the present system, we will be overcoming it by making the system wireless using Bluetooth.

Chapter 12

Conclusion

As stated before, the electricity needs to be saved because it cannot be created. Our project is an automatic independent system which helps us in saving a lot amount of energy. We don't need to put any effort because the system is smart enough to work on it's own.

References

[1] www.atmel.com

[2] www.beyondlogic.org

[3] www.wikipedia.org

[4] www.howstuffworks.com

[5] www.alldatasheets.com