A PROJECT REPORT ON

IOT BASED SMART ENERGY METER MONITORING WITH THEFT DETECTION

IN PARTIAL FULFILLMENT OF

BACHELOR OF TECHNOLOGY

IN

ELECTRICAL AND ELECTRONICS ENGINEERING

Submitted by

1.SURYA SAAGAR KARAN	**AJU/181392**
2.AKHILESH MISHRA	**AJU/181372**
3.KESHAV SHUKLA	**AJU/181394**

Under the guidance of

Mr. Kumaresh Pal
Assistant professor
Department of Engineering
ARKA JAIN UNIVERSITY,JHARKHAND

ARKA JAIN University, Jharkhand
2021-2022

ARKA JAIN University
Jharkhand

CERTIFICATE

DATE: 10-05-22

This is to certify that project entitled *"IOT BASED SMART ENERGY METER MONITORING WITH THEFT DETECTION"* has been submitted to the Department of Electrical and Electronics Engineering, ARKA JAIN University, Jharkhand for the fulfilment of the requirement for the award of degree of "*Bachelor of Technology in Electrical & Electronics Engineering*" by following student of final year BTECH (Electrical and Electronics Engineering).

SURYA SAAGAR KARN **AJU/181392**

AKHILESH MISHRA **AJU/181372**

KESHAV SHUKLA **AJU/181394**

Mr. Kumaresh Pal

(Project Guide)

Mr. Ashwini Kumar

(Asst. Dean)

Asst. Dean
School of Engineering & IT
ARKA JAIN University

DECLARATION BY THE CANDIDATE

I hereby declare that the project entitled ***"IOT BASED SMART ENERGY METER MONITORING WITH THEFT DETECTION"*** **submitted by us to ARKA JAIN University, Jharkhand in partial fulfillment of the requirement for the award of the degree of "*Bachelor of Technology in Electrical & Electronics Engineering*" is a record of bona fide project work carried out by us (project group member) under the guidance of Mr. Kumaresh Pal (Asst. Prof., E.E.E. dept.). I further declare that the work reported in this project has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or B.TECH in this university or any other institute or university.**

We will be solely responsible if any kind of plagiarism is found.

Date:- 10-05-22

ARKA JAIN UNIVERSITY

Surya Saagar Karn
10.5.22

MR. SURYA SAAGAR KARN

AJU/181392

Akhilesh Mishra
10-05-22

MR. AKHILESH MISHRA

AJU/181372

Keshav Shukla
10-05-22

MR. KESHAV SHUKLA

AJU/181394

ACKNOWLEDGEMENT

We like to share our sincere gratitude to all those who help us in completion of this project. During the work we faced many challenges due to our lack of knowledge and experience, but these people help us to get over from all the difficulties and in final completion of our idea to a shaped sculpture.

We would like to thank Mr. Ashwini Kumar (Asst. Dean of Engineering & IT Department) for his governance and guidance, because of which our whole team was able to learn the minute aspects of a project work.

We would also like to thank our Project Guide, Mr. Kumaresh Pal (Asst. Prof., E.E.E. dept.) for his continuous help and monitoring during the project work.

We would like to thank the management of ARKA JAIN University for providing us such an opportunity to learn from their experiences.
All our team members are also thankful to all the faculties and staff of Department of Electrical & Electronics Engineering, AJU, Jharkhand, for their help and support towards this project and our team.

We are also thankful to our classmates and most of all to our parents who have inspired to face all the challenges and will all the hurdles in life.

DATE:	**SURYA SAAGAR KARAN**	**AJU/181392**
ARKA JAIN UNIVERSITY	**AKHILESH MISHRA**	**AJU/181372**
	KESHAV SHUKLA	**AJU/181394**

ABSTRACT

Energy crisis is one of the major problems that the world faces today. The energy crisis can be reduced to a certain extent by properly monitoring our energy consumption and avoiding energy wastage. Nowadays people face many problems like power theft. Power theft may be a measure crime and it also directly affects the economy of our country. This system will find energy theft easily.

This IOT electricity meter is consisting of Atmega 328 microcontroller with a WIFI module for IOT connection and GSM module for mobile connection, on which users will receive information via SMS. This smart electricity meter also consists of a current sensor that sends the current reading to the microcontroller.

We have to connect cell phones with the system via SMS which will help to configure with the system. In case of an emergency, the information will be shared on the configured number. We have to set costs for the unit and for which we have four buttons. With the help of buttons, we can set costs for the unit. As we start the system, it shows reading on the IOT screen. Reading will be changed with respect to time. In the case of energy theft, the theft will be caught and displayed on the IOT screen. Even the information will be received through SMS on the configured number. After receiving the alert, the operator can switch off the system using IOT to avoid theft. It also shares turn off the message of the system on the cell phone.

Keywords: Atmega 328 Microcontroller, Wifi Module, GSM Module ,Smart Electricity Meter

CONTENT

List of Symbols

UNIX	UNiplexed Information computing system
OS	Operating System
GFSK	Gaussian Frequency Shift Keying Modulation
O-QPSK	Offset Quadrature phase Shift Keying
BPSK	Binary Phase Shift Keying
MR-FSK	Multi- Resonant Frequency Shift Keying
MR-OFDM	Multi- Regional Orthogonal Frequency Division Multiplexing
CCS	Chirp Spread Spectrum
EEPROM	Erasable Programmable Read Only Memory
SRAM	Static Random Access Memory
ISP	Internet Service Provider
UART	Universal Asynchronous Reciever Transmitter
USART	Universal Synchronous Asynchronous Reciever Transmitter

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 Understanding Internet of Things

Everyone is eyeing the next big thing after the .com boom which will make riches. World has never being the same after advent of the internet. Investment gurus and statisticians may have many proposals to make but one thing is for sure, the next big move which will shape the century will depend on internet and embedded technology. That is, in other words internet of things definition is what interests major players now. What we do, how we do and when we do is never going to be the same when the physical environment around us gets lively and starts communicating.

The Internet of Things (IoT) is here and is becoming an increasing topic of interest among technology giants and business communities. The hype is not baseless as there are enough evidences to support the success of "Internet of Things" in the coming years. According to a report by Gartner there will be 30% increase in the number of connected devices in 2016 as compared to 2015 with 6.4 billion IoT devices entering the realm of internet of things. The number is further expected to increase to 26 billion by 2020.

So one might simply ask "What is Internet of Things" and how it is going to impact our lives and career opportunities. There is a lot of complicated technology and terminologies at work in the IoT world but here I will try to keep things simple to explain the concept of Internet of Things easily.

1.1.1 Internet of Things Definition

IoT is simply the network of interconnected things/devices which are embedded with sensors, software, network connectivity and necessary electronics that enables them to collect and exchange data making them responsive.

More than a concept Internet of Things is essentially an architectural framework which allows integration and data exchange between the physical world and computer systems over existing network infrastructure.

1.1.2 Important Internet of Things Components

Many people mistakenly think of IoT as an independent technology. Interestingly internet of things is being enabled by the presence of other independent technologies which make fundamental components of IoT. The fundamental components that make internet of things a reality are:

- **Hardware-Making physical objects responsive and giving them capability to retrieve data and respond to instructions**
- **Software-Enabling the data collection, storage, processing, manipulating and instructing**
- **Communication Infrastructure-Most important of all is the communication infrastructure which consists of protocols and technologies which enable two physical objects to exchange data**

Why Internet of Things/IoT will be successful in the coming years?

As the telecommunication sector is becoming more extensive and efficient, broadband internet is widely available. With technological advancement it is now much cheaper to produce necessary sensors with built-in wifi capabilities making connecting devices less costly.

Most important, the smart phone usage has surpassed all the predicted limits and telecommunication sector is already working on its toes to keep their customers satisfied by improving their infrastructure. As IoT devices need no separate communication than the existing one building IoT tech is very cheap and highly achievable

1.1.3 HISTORY of IoT

The main concept of a network of smart devices was discussed as early as 1982, with a modified Coca-Cola vending machine at Carnegie Mellon University becoming the first ARPANET-connected appliance, able to report its inventory and whether newly loaded drinks were cold or not. Mark Weiser's 1991 paper on ubiquitous computing, "The Computer of the 21st Century", as well as academic venues such as UbiComp and PerCom produced the contemporary vision of the IOT. In 1994, Reza Raji described the concept in IEEE Spectrum as [moving] small packets of data to a large set of nodes, so as to integrate and automate everything from home appliances to entire factories. Between 1993 and 1997, several companies proposed solutions like Microsoft's at Work or Novell's NEST. The field gained momentum when Bill Joy envisioned device-to-device communication as a part of his "Six Webs" framework, presented at the World Economic Forum at Davos in 1999.

The concept of the "Internet of things" and the term itself, first appeared in a speech by Peter T. Lewis, to the Congressional Black Caucus Foundation 15th Annual Legislative Weekend in Washington, D.C, published in September 1985. According to Lewis, "The Internet of Things, or IoT, is the integration of people, processes and technology with connectable devices and sensors to enable remote monitoring, status, manipulation and evaluation of trends of such devices."

The term "Internet of things" was coined independently by Kevin Ashton of Procter & Gamble, later MIT's Auto-ID Center, in 1999 though he prefers the phrase "Internet *for* things". At that point, he viewed radio-frequency identification (RFID) as essential to the Internet of things, which would allow computers to manage all individual things. The main theme of the Internet of things is to embed short-range mobile transceivers in various gadgets and daily necessities to enable new forms of communication between people and things, and between things themselves.

Defining the Internet of things as "simply the point in time when more 'things or objects' were connected to the Internet than people", Cisco Systems estimated that the IoT was "born" between 2008 and 2009, with the things/people ratio growing from 0.08 in 2003 to 1.84 in 2010.

1.1.4 Internet of Things Infographic

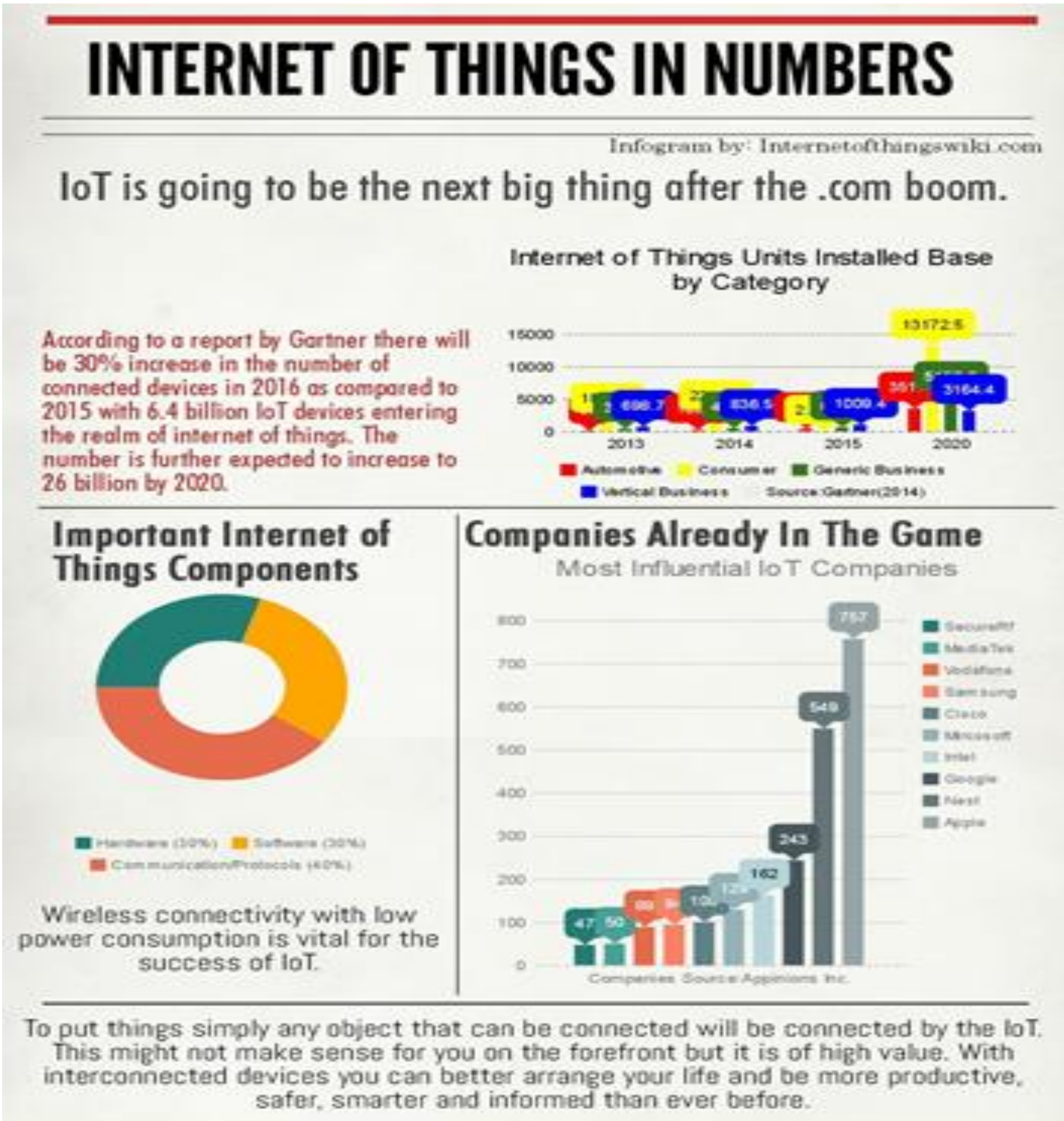


FIG 1.1 The Impact of Internet of Things on You

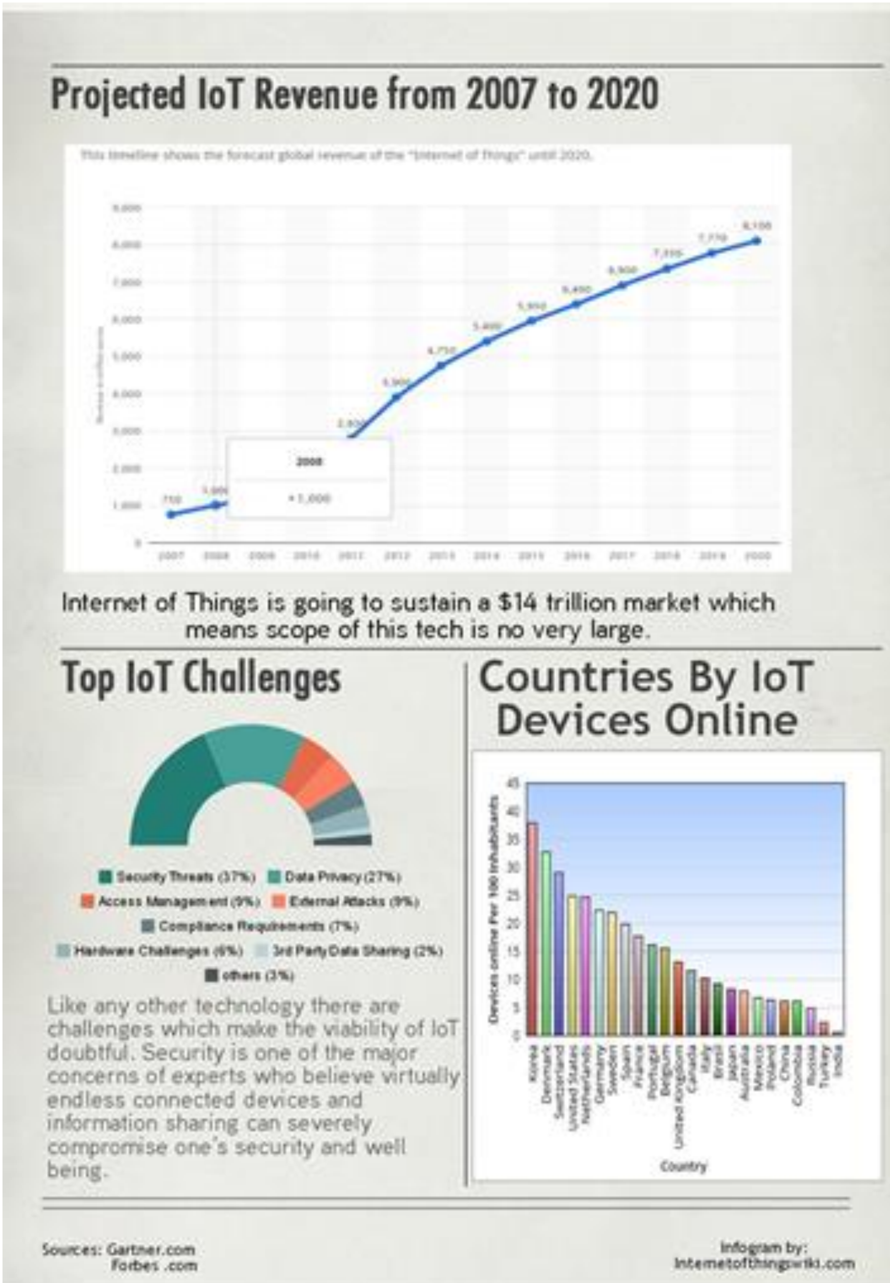


FIG 1.2

To put things simply any object that can be connected will be connected by the IoT. This might not make sense for you on the forefront but it is of high value. With interconnected devices you can better arrange your life and be more productive, safer, smarter and informed than ever before.

For instance how easy it will be for you to start your day if your alarm clock is not only able to wake you up but also able to communicate with your brewer to inform it that you are awake at the same time notifies your geezer to start water heating. Or you wearable wrist health band keeps track of your vitals to inform you when you are most productive during the day. These are just few examples but applications of internet of things are numerous.

On large scale transportation, healthcare, defence, environment monitoring, manufacturing and every other field you can imagine of can be benefited from IoT. It is very to conceive the whole application domain of internet of things at the moment but you can clearly understand why it is such an interesting and hot topic at the moment.

1.1.5 The Challenges of Internet of Things

Like any other technology there are challenges which make the viability of IoT doubtful. Security is one of the major concerns of experts who believe virtually endless connected devices and information sharing can severely compromise one's security and well being. Unlike other hacking episodes which compromise online data and privacy with IoT devices can open gateway for an entire network to be hacked.

One such flaw is well presented by Andy Greenberg on wired.com where he works with hackers to remotely kill his Jeep on the highway. Another very relevant example is provided by W. David Stephenson in his post Amazon Echo: is it the smart home Trojan Horse? You can estimate the amount of personal and private data the connected devices will be producing once they are on a network. The major challenge for IoT tech companies is to figure out how the communication in the internet of things realm can be made truly secure.

IoT wireless connectivity performance standards and characteristics

Name	Specification	Modulation	Frequency (MHz)	Bandwidth (MHz)	Range (m)
Bluetooth	*Bluetooth* SIG	GFSK, D8PSK	2400	1	50
ZigBee	IEEE 802.15.4	O-QPSK, BPSK	780, 868, 915, 920, 2450	2	10
WiSUN	IEEE 802.15.4g	MR-FSK, MR-OFDM	920	0.2-1.2	1,000
LoRaWAN	LoRa Alliance	GFSK, CCS	169, 433, 470, 868, 915	0.5	10,000
Z-Wave	ITU-T G9959	FSK, GFSK	868, 915, 920	0.2	100
HaLow	802.11ah	OFDM	779, 868, 915, 920	1, 2, 4, 8, 16	1,000
DSRC / WAVE	802.11p	OFDM	5800, 5900	5, 10, 20	1,000
Cat-NB2 (NB-IoT)	3GPP Rel-13	BPSK, QPSK, 16-QAM	GSM / LTE bands	0.18	1,000
Cat-M1	3GPP Rel-13	OFDM	LTE bands	1.4	1,000
C-V2X	3GPP Rel-14, Rel-16	OFDM	Bands 3, 7, 8, 39, 41, 47	10	2,000

TABLE 1.1

1.1.6 What to Expect

Any object that can be connected will be connected by the IoT.

Internet of Things is truly a game changing concept and whatever challenges may be present, there will be rapid growth in the number of connected devices. There is still need for us to understand concepts of IoT and the flaws that are underlying. As more and more devices start to connect the need will force technocrats to come up with most advanced concepts and methods to ensure growth of this technology.

By the time what we can do best is to educate ourselves on various IoT technologies and keep experimenting with new stuff.

1.1.7 APPLICATIONS

The extensive set of applications for IoT devices is often divided into consumer, commercial, industrial, and infrastructure spaces.

- Consumer applications

A growing portion of IoT devices are created for consumer use, including connected vehicles, home automation, wearable technology, connected health, and appliances with remote monitoring capabilities.

  - Smart home

IoT devices are a part of the larger concept of home automation, which can include lighting, heating and air conditioning, media and security systems and camera systems. Long-term benefits could include energy savings by automatically ensuring lights and electronics are turned off or by making the residents in the home aware of usage.

FIG 1.3 A Smart Toilet Seat that measures blood pressure, weight, pulse and oxygen levels.

A smart home or automated home could be based on a platform or hubs that control smart devices and appliances. For instance, using Apple's Home Kit, manufacturers can have their home products and accessories controlled by an application in iOS devices such as the iPhone and the Apple Watch. This could be a dedicated app or iOS native applications such as Siri. This can be demonstrated in the case of Lenovo's Smart Home Essentials, which is a line of smart home devices that are controlled through Apple's Home app or Siri without the need for a Wi-Fi bridge. There are also dedicated smart home hubs that are offered as standalone platforms to connect different smart home products and these include the Amazon Echo, Google Home, Apple's Home Pod, and Samsung's Smart Things Hub. In addition to the commercial systems, there are many non-proprietary, open source ecosystems; including Home Assistant, Open HAB and Domoticz.

- Elder care

One key application of a smart home is to provide assistance for those with disabilities and elderly individuals. These home systems use assistive technology to accommodate an owner's specific disabilities. Voice control can assist users with sight and mobility limitations while alert systems can be connected directly to cochlear implants worn by hearing-impaired users. They can also be equipped with additional safety features. These features can include sensors that monitor for medical emergencies such as falls or seizures. Smart home technology applied in this way can provide users with more freedom and a higher quality of life.

The term "Enterprise IoT" refers to devices used in business and corporate settings. By 2019, it is estimated that the EIoT will account for 9.1 billion devices.

- **Organizational applications**

- Medical and healthcare

The Internet of Medical Things (IoMT) is an application of the IoT for medical and health related purposes, data collection and analysis for research, and monitoring. The IoMT has been referenced as "Smart Healthcare", as the technology for creating a digitized healthcare system, connecting available medical resources and healthcare services.

IoT devices can be used to enable remote health monitoring and emergency notification systems. These health monitoring devices can range from blood pressure and heart rate monitors to advanced devices capable of monitoring specialized implants, such as

pacemakers, Fitbit electronic wristbands, or advanced hearing aids. Some hospitals have begun implementing "smart beds" that can detect when they are occupied and when a patient is attempting to get up. It can also adjust itself to ensure appropriate pressure and support is applied to the patient without the manual interaction of nurses. A 2015 Goldman Sachs report indicated that healthcare IoT devices "can save the United States more than $300 billion in annual healthcare expenditures by increasing revenue and decreasing cost."

  - **Transportation**
  - **V2X communications**
  - **Building and home automation**
- **Industrial applications**
  - **Manufacturing**
  - **Agriculture**
  - **Maritime**
- **Infrastructure applications**
  - **Metropolitan scale deployments**
  - **Energy management**
  - **Environmental monitoring**
- **Military applications**
  - **Internet of Battlefield Things**
  - **Ocean of Things**
- **Product digitalization**

1.2 INTRODUCTION TO EMBEDDED SYSTEMS

1.2.1 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. Examples Small controllers and devices in our everyday life like Washing Machine,Microwave Ovens, where they are embedded in

FIG 1.4 An Embedded System on a plug-in card with processor, memory, power supply, and external interfaces

1.2.2 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.**
  - **Response–Our system may need to react to events quickly**
  - **Testability–Setting up equipment to test embedded software can be difficult**
  - **Debuggability–Without 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.**

1.2.3 History of Embedded system

Background

The origins of the microprocessor and the microcontroller can be traced back to the MOS integrated circuit, which is an integrated circuit chip fabricated from MOSFETs (metal-oxide-semiconductor field-effect transistors) and was developed in the early 1960s. By 1964, MOS chips had reached higher transistor density and lower manufacturing costs than bipolar chips. MOS chips further increased in complexity at a rate predicted by Moore's law, leading to large-scale integration (LSI) with hundreds of transistors on a single MOS chip by the late 1960s. The application of MOS LSI chips to computing was the basis for the first microprocessors, as engineers began recognizing that a complete computer processor system could be contained on several MOS LSI chips.

The first multi-chip microprocessors, the Four-Phase Systems AL1 in 1969 and the Garrett AiResearch MP944 in 1970, were developed with multiple MOS LSI chips. The first single-chip microprocessor was the Intel 4004, released in 1971. It was developed by Federico Faggin, using his silicon-gate MOS technology, along with Intel engineers Marcian Hoff and Stan Mazor, and Busicom engineer Masatoshi Shima.

Development

One of the first recognizably modern embedded systems was the Apollo Guidance Computer, developed ca. 1965 by Charles Stark Draper at the MIT Instrumentation Laboratory. At the project's inception, the Apollo guidance computer was considered the riskiest item in the Apollo project as it employed the then newly developed monolithic integrated circuits to reduce the computer's size and weight.

An early mass-produced embedded system was the Autonetics D-17 guidance computer for the Minuteman missile, released in 1961. When the Minuteman II went into production in 1966, the D-17 was replaced with a new computer that represented the first high-volume use of integrated circuits.

Since these early applications in the 1960s, embedded systems have come down in price and there has been a dramatic rise in processing power and functionality. An early microprocessor, the Intel 4004 (released in 1971), was designed for calculators and other small systems but still required external memory and support chips. By the early 1980s, memory, input and output system components had been integrated into the same chip as the processor forming a microcontroller. Microcontrollers find applications where a general-purpose computer would be too costly. As the cost of microprocessors and microcontrollers fell the prevalence of embedded systems increased.

Today, a comparatively low-cost microcontroller may be programmed to fulfill the same role as a large number of separate components. With microcontrollers, it became feasible to replace, even in consumer products, expensive knob-based analog components such as potentiometers and variable capacitors with up/down buttons or knobs read out by a microprocessor. Although in this context an embedded system is usually more complex than a traditional solution, most of the complexity is contained within the microcontroller itself. Very few additional components may be needed and most of the design effort is in the software. Software prototype and test can be quicker compared with the design and construction of a new circuit not using an embedded processor.

1.2.4 APPLICATIONS

1) **Military and aerospace embedded software applications**
2) **Communication Applications**
3) **Industrial automation and process control software**
4) **Mastering the complexity of applications.**
5) **Reduction of product design time.**
6) **Real time processing of ever increasing amounts of data.**
7) **Intelligent, autonomous sensors.**

1.2.5 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**

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

LITERATURE REVIEW

2.1 POWER THEFT IDENTIFICATION

Today the most challenging problems are the power theft. In previous days, power thefts are very complicate to identify manually. In this project we also added the theft control system; this system will identify that which consumer is pilfering the supply from other consumer furtively.

Once the robbery connection is identified with the help of that system, then it will be informed to the authorized consumers through SMS.

As said earlier LCD is used for displaying purpose. It show how much units consumed by the users, bill amount with billing date and due date for that bill amount.

This system will be advantageous to people because it will reduce the penalty amount which was given by the government if the bill is not paid on or before the due date.

2.2 THEFT CONTROL

- **There are two types of controlling process in theft control.**

- **IR sensor is the first process in theft control and it is used for identifying the tapering of seal.**

- **Thefts of energy meter that don’t work properly are one of the major causes of commercial losses.**

• Checking the billing procedure of the utility can be the first step to reduce commercial losses.

• To do that the consumer data base must be always updated, reflecting any change verified in the field with the smallest time delay.

• This data base is important not only for auditing the billing. Once the robbery connection is identified with the help of that system, then it will be informed to the authorized consumers through SMS.

2.3 ADVANTAGES

- **Fast operation.**
- **Operating principle is simple.**
- **Easy to handle.**
- **Time Saving.**
- **Compact size and portable.**
- **Easy to move from one place to another.**
- **Non-skilled person also operate this machine**

2.4 DISADVANTAGES

- **Cost of the equipment is high.**
- **Environment consideration.**
- **Regular maintenance is required.**

2.5 APPLICATIONS

- **For Commercial use.**
- **For home purpose.**

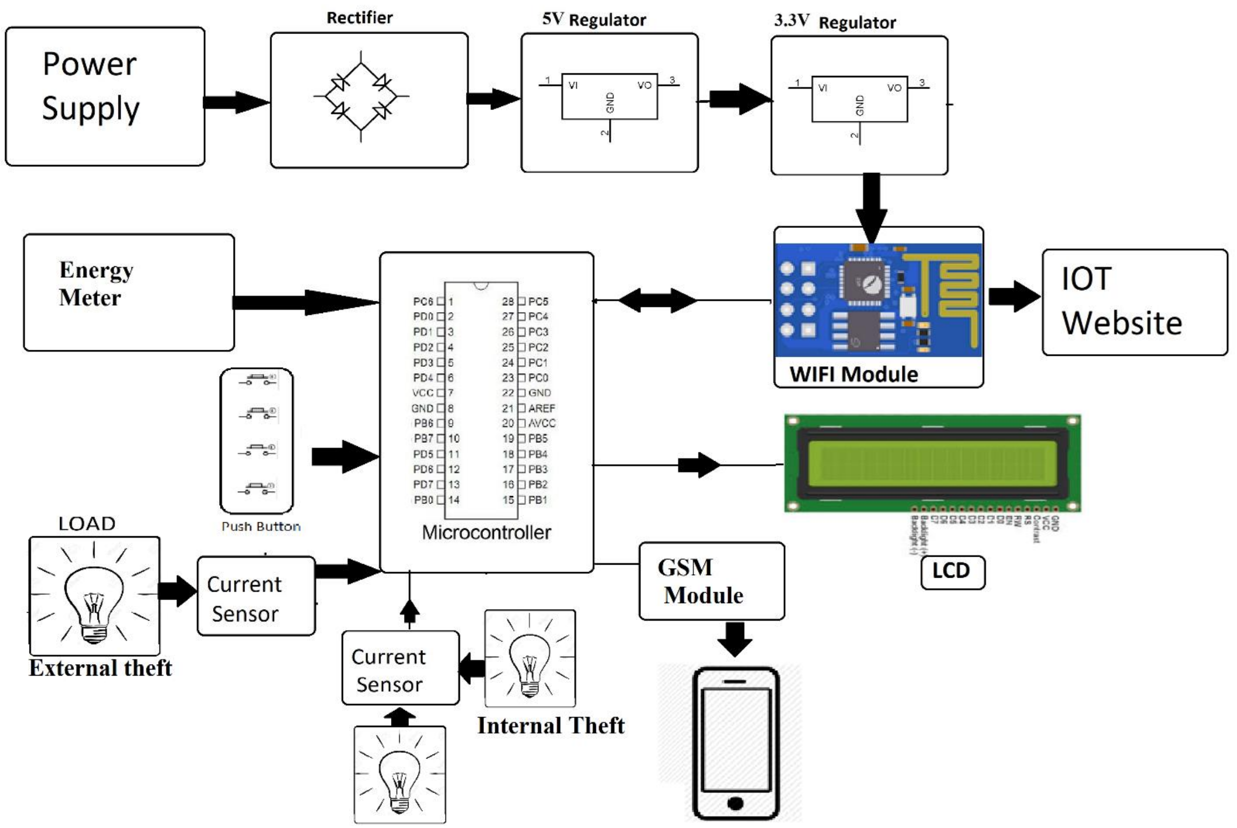
2.6 SAFETY PRECAUTIONS

- **Do not press any switch without knowing its working.**
- **Do not open any connection when power supply is on.**
- **Handle its carefully.**

CHAPTER 3

DESIGN PROCEDURES

3.1 BLOCK DIAGRAM



3.2 HARDWARE REQUIREMENTS

HARDWARE COMPONENTS:

- **Energy Meter**
- **Transformer/Adapter**
- **Atmega 328 Microcontroller**
- **Wifi Modem ESP8266**
- **LCD Display**
- **LED’s**
- **Transformer**
- **Resistors**
- **Capacitors**
- **Diodes**

3.3 VOLTAGE REGULATOR 7805

Features

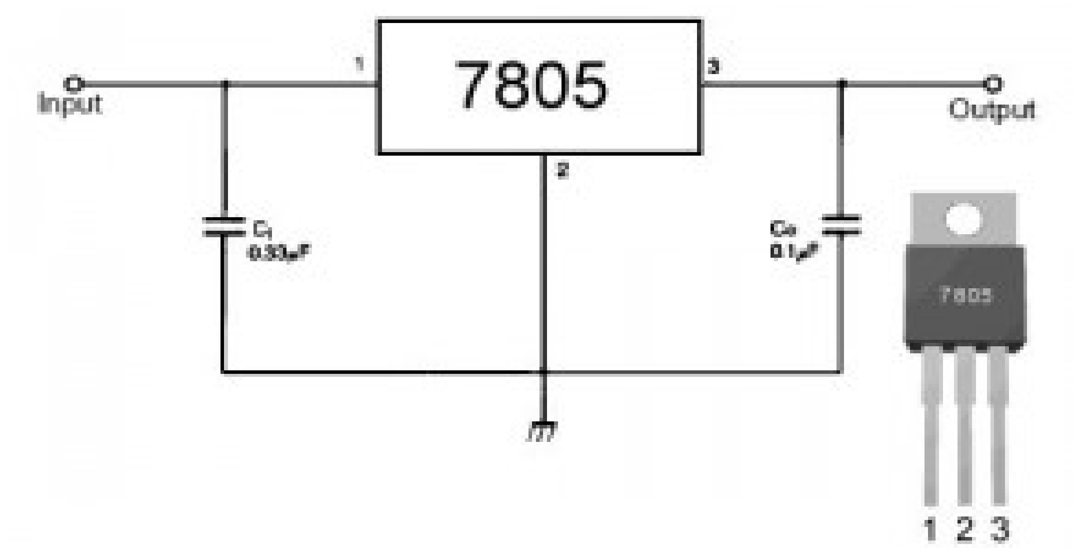
• Output Current up to 1A.

• Output 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

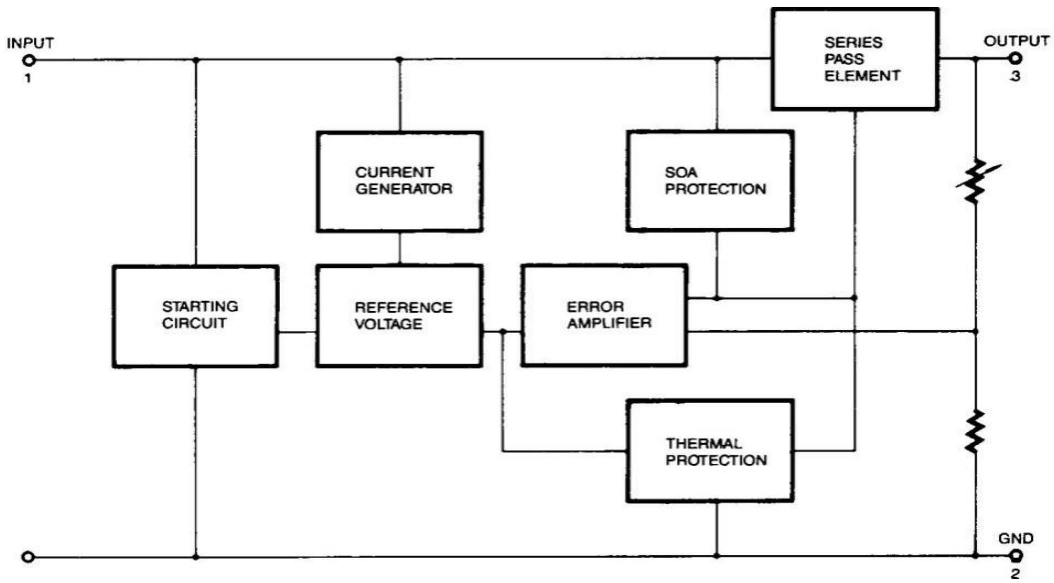


FIG 3.3: BLOCK DIAGRAM OF VOLTAGE REGULATOR

Absolute Maximum Ratings

Parameter	Symbol	Value	Unit
Input Voltage (for V_O = 5V to 18V) (for V_O = 24V)	V_I V_I	35 40	V V
Thermal Resistance Junction-Cases (TO-220)	$R_{\theta JC}$	5	°C/W
Thermal Resistance Junction-Air (TO-220)	$R_{\theta JA}$	65	°C/W
Operating Temperature Range (KA78XX/A/R)	T_{OPR}	0 ~ +125	°C
Storage Temperature Range	T_{STG}	-65 ~ +150	°C

TABLE 3.1: RATINGS OF THE VOLTAGE REGULATOR

3.4 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.

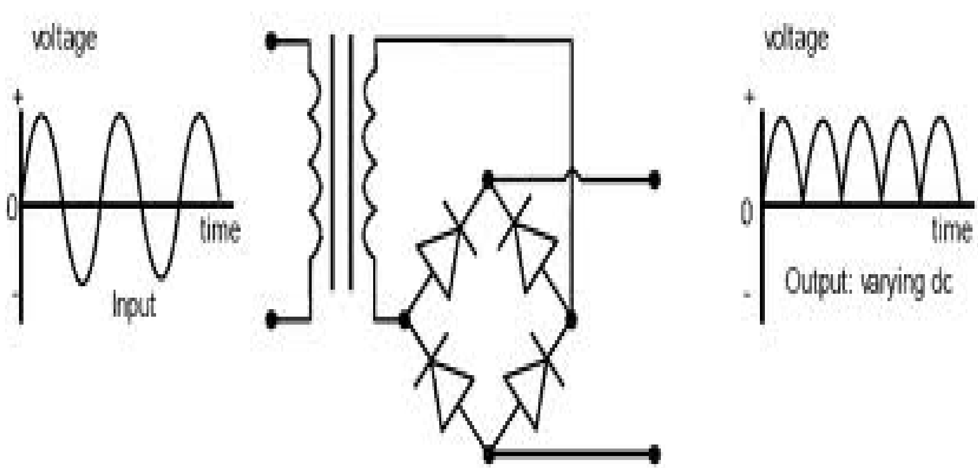


FIG 3.4 Rectifier

3.5 FILTER

Capacitive filter is used in this project. It removes the ripples from the output of rectifier and smoothens 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 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.

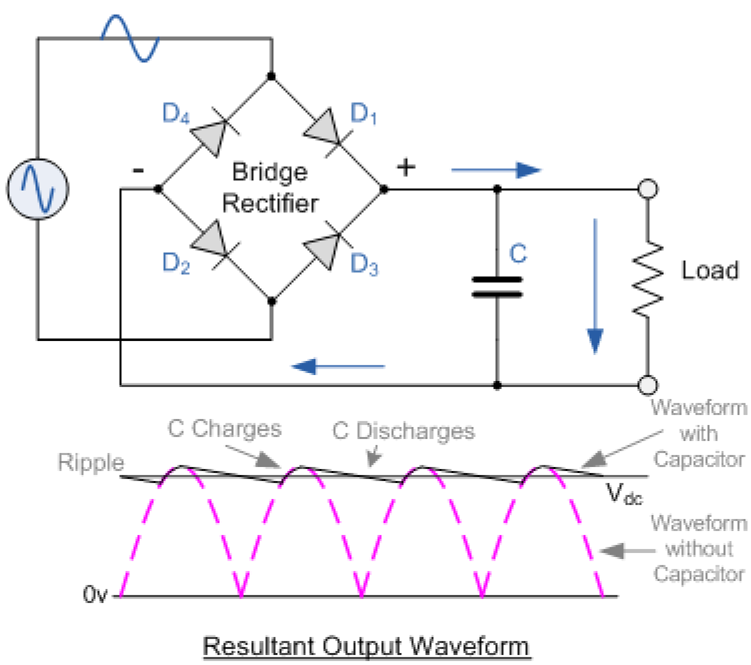
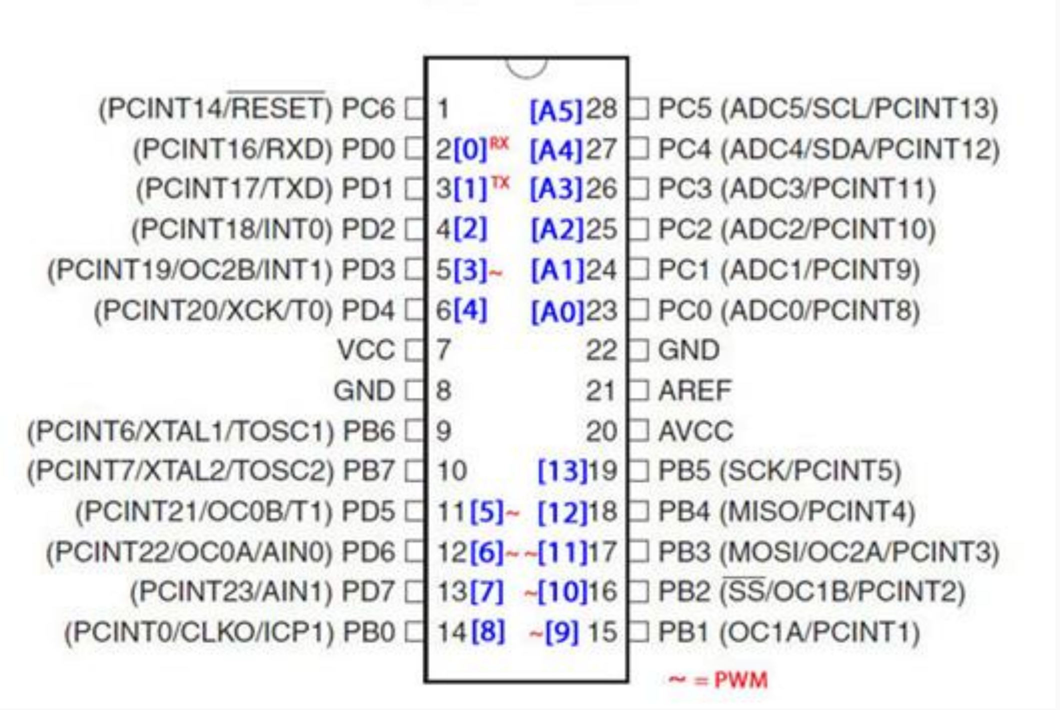


FIG 3.5

3.6 ATMEGA328

FIG 3.6



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.

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 85°C and 100 years at 25°C**
  - **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 Include**
  - **2 x 8-bit Timers/Counters each with independent prescaler and compare modes**
  - **A single 16-bit Timer/Counter with an idependent 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:**
    - 40°C to 85°C
- **Speed Grades:**
    - 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 25°C:**
    - Active Mode: 0.3 mA
    - Power-down Mode: 0.1 µA
    - Power-save Mode: 0.8 µA (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**

In our days, there have been many advancement in the field of Electronics and many cutting edge technologies are being developed every day, but still 8 bit microcontrollers have its own role in the digital electronics market dominated by 16-32 & 64 bit digital devices. Although powerful microcontrollers with higher processing capabilities exist in the market, 8bit microcontrollers still hold its value because of their easy-to-understand-operation, very much high popularity, ability to simplify a digital circuit, low cost compared to features offered, addition of many new features in a single IC and interest of manufacturers and consumers.

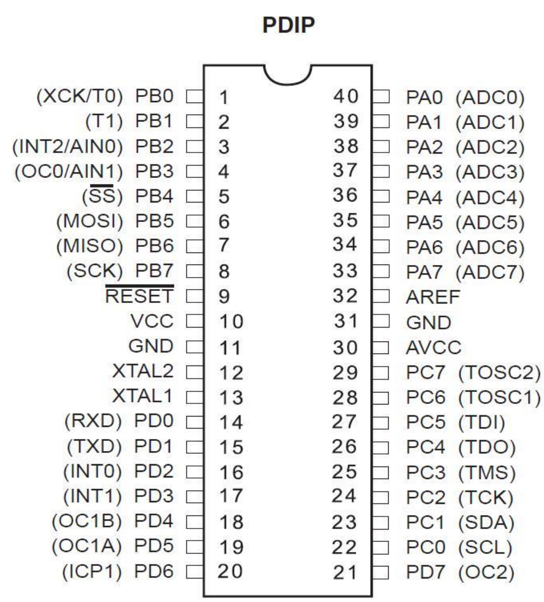
Today's microcontrollers are much different from what it were in the initial stage, and the number of manufacturers are much more in count than it was a decade or two ago. At present some of the major manufacturers are Microchip (publication: PIC microcontrollers), Atmel (publication: AVR microcontrollers), Hitachi, Phillips, Maxim, NXP, Intel etc. Our interest is upon ATmega32. It belongs to Atmel's AVR series micro controller family. Let's see the features.
PIN count: Atmega32 has got 40 pins. Two for Power (pin no.10: +5v, pin no. 11: ground), two for oscillator (pin 12, 13), one for reset (pin 9), three for providing necessary power and reference voltage to its internal ADC, and 32 (4×8) I/O pins.

About I/O pins: ATmega32 is capable of handling analogue inputs. Port A can be used as either DIGITAL I/O Lines or each individual pin can be used as a single input channel to the internal ADC of ATmega32, plus a pair of pins AREF, AVCC & GND (refer to ATmega32 datasheet) together can make an ADC channel.
No pins can perform and serve for two purposes (for an example: Port A pins cannot work as a Digital I/O pin while the Internal ADC is activated) at the same time. It's the programmers responsibility to resolve the conflict in the circuitry and the program. Programmers are advised to have a look to the priority tables and the internal configuration from the datasheet.

Digital I/O pins: ATmega32 has 32 pins (4portsx8pins) configurable as Digital I/O pins.
Timers: 3 Inbuilt timer/counters, two 8 bit (timer0, timer2) and one 16 bit (timer1).
ADC: It has one successive approximation type ADC in which total 8 single channels are selectable. They can also be used as 7 (for TQFP packages) or 2 (for DIP packages) differential channels. Reference is selectable, either an external reference can be used or the internal 2.56V reference can be brought into action. There external reference can be connected to the AREF pin.
Communication Options: ATmega32 has three data transfer modules embedded in it. They are

- **Two Wire Interface**
- **USART**
- **Serial Peripheral Interface**

Fig 3.7



Analog comparator: On-chip analog comparator is available. An interrupt is assigned for different comparison result obtained from the inputs.
External Interrupt: 3 External interrupt is accepted. Interrupt sense is configurable.
Memory: It has 32K bytes of In-System Self-programmable Flash program memory, 1024 Bytes EEPROM, 2K bytes Internal SRAM. Write/Erase Cycles: 10,000 Flash / 100,000 EEPROM.
Clock: It can run at a frequency from 1 to 16 MHz. Frequency can be obtained from external Quartz Crystal, Ceramic crystal or an R-C network. Internal calibrated RC oscillator can also be used.
More Features: Up to 16 MIPS throughput at 16MHz. Most of the instruction executes in a single cycle. Two cycle on-chip multiplication. 32 × 8 General Purpose Working Registers
Debug: JTAG boundary scan facilitates on chip debug.
Programming: Atmega32 can be programmed either by In-System Programming via Serial peripheral interface or by Parallel programming. Programming via JTAG interface is also possible. Programmer must ensure that SPI programming and JTAG are not be disabled using fuse bits; if the programming is supposed to be done using SPI or JTAG.

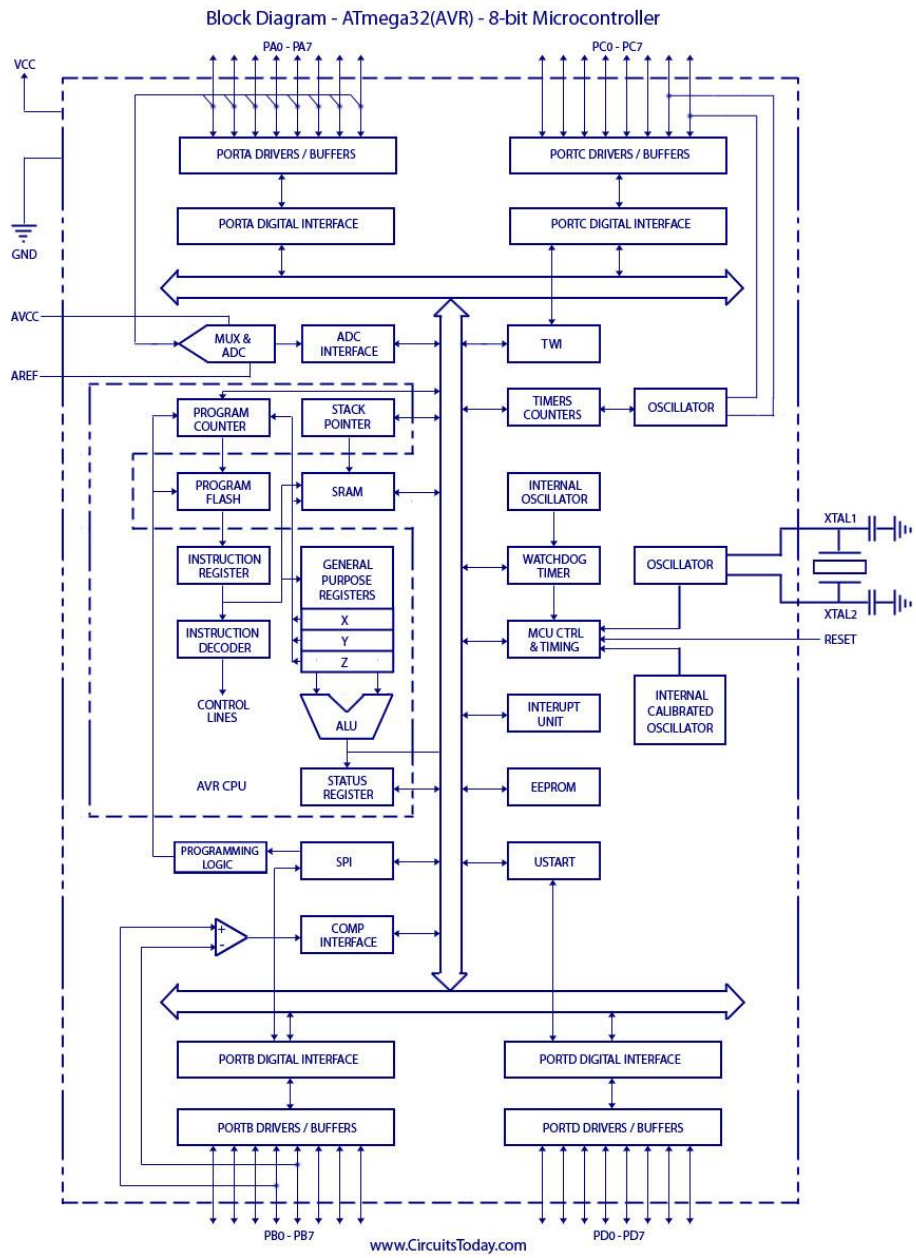


Fig 3.8

3.7 LCD DISPLAY

Description:

This is the example for the Parallel Port. This example doesn't use the Bi-directional feature found on newer ports, thus it should work with most, if not 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.

Pros:

- **Very compact and light**
- **Low power consumption**
- **No geometric distortion**
- **Little or no flicker depending on backlight technology**
- **Not affected by screen burn-in**
- **No high voltage or other hazards present during repair/service**
- **Can be made in almost any size or shape**
- **No theoretical resolution limit**

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.

FIG 3.9: LCD

3.7.1 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).

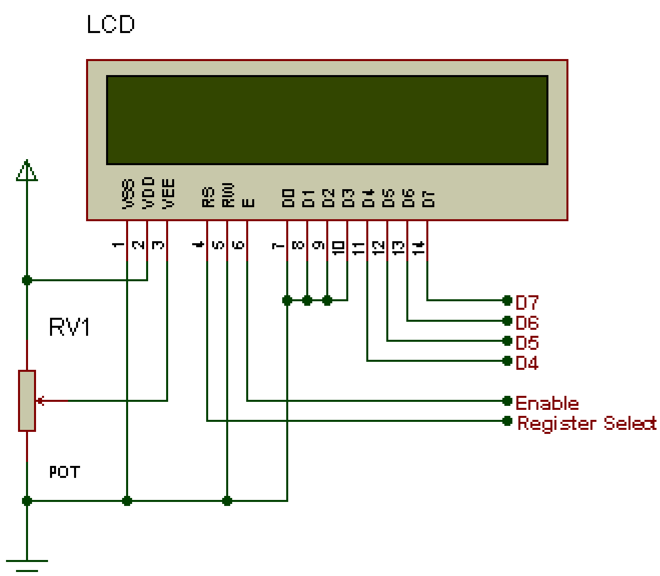


FIG 3.10

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.

3.8 ENERGY METER

An energy or electric meter is a device that measures the amount of electrical energy consumed by a residence, business, or an electrically-powered device.

FIG 3.11: Energy Meter

Electric meters are typically calibrated in billing units, the most common one being the kilowatt hour. Periodic readings of electric meters establish billing cycles and energy used during a cycle.

In settings when energy savings during certain periods are desired, meters may measure demand, the maximum use of power in some interval. In some areas, the electric rates are higher during certain times of day, to encourage reduction in use. Also, in some areas meters have relays to turn off nonessential equipment.

3.8.1 Unit of measurement

The most common unit of measurement on the electricity meter is the *kilowatt hour,* which is equal to the amount of energy used by a load of one kilowatt over a period of one hour, or 3,600,000 joules.

Demand is normally measured in watts, but averaged over a period, most often a quarter or half hour.

Reactive power is measured in "Volt-amperes reactive", (varh) in kilovar-hours. By convention, a "lagging" or inductive load, such as a motor, will have positive reactive power. A "leading", or capacitive load, will have negative reactive power.

Volt-amperes measures all power passed through a distribution network, including reactive and actual. This is equal to the product of root-mean-square volts and amperes.

Meters which measured the amount of charge (coulombs) used, known as ampere-hour meters, were used in the early days of electrification. These were dependent upon the supply voltage remaining constant for accurate measurement of energy usage, which was not a likely circumstance with most supplies.

Some meters measured only the length of time for which charge flowed, with no measurement of the magnitude of voltage or current is being made. These were only suited for constant-load applications.

3.8.2 Electromechanical meters

The most common type of electricity meter is the electromechanical induction watt-hour meter. The electromechanical induction meter operates by counting the revolutions of an aluminium disc which is made to rotate at a speed proportional to the power. The number of revolutions is thus proportional to the energy usage. It consumes a small amount of power, typically around 2 watts.

The metallic disc is acted upon by two coils. One coil is connected in such a way that it produces a magnetic flux in proportion to the voltage and the other produces a magnetic flux in proportion to the current. The field of the voltage coil is delayed by 90 degrees using a lag coil.[17] This produces eddy currents in the disc and the effect is such that a force is exerted on the disc in proportion to the product of the instantaneous current and voltage. A permanent magnet exerts an opposing force proportional to the speed of rotation of the disc. The equilibrium between these two opposing forces results in the disc rotating at a speed proportional to the power being used. The disc drives a register mechanism which integrates the speed of the disc over time by counting revolutions, much like the odometer in a car, in order to render a measurement of the total energy used over a period of time.

The type of meter described above is used on a single-phaseAC supply. Different phase configurations use additional voltage and current coils.

3.8.3 Electro Mechanical Induction motor Properties:

Application:	**Type SY1029 single-phase watt-hour meter is induction meter, which is applicable to measure the rated frequency 50Hz and power loss in electrified wire netting. The meter has novel design, rational structure and features of high overload, low power loss and long life etc. All technical targets are completely conformed to National Standard GB/15283-94 and International Standard IEC60521.**
Structure and Features:	**1. Sub-closed electromagnetic core, the die-casting frame is made of alloy aluminum, assure magnetism stable and reliable.** **2. Meter bearing has three kinds, may select dual jewel, magnetic thrust or magnetic float.** **3. May select 5+1 digits or 4+1 digits register.**

4. May select 5 kinds of cases: bakelite, PC, ABS plastic, glass and Aluminum.

Technical Parameters:

Rate Voltage: 220V
Rate Current: 1.5(6),2.5(10),3(12),5(20), 10(30),10(40),15(60),20(60), 20(80),30(90),30(100)
Overload Capacity: 200%~400%Ib
Starting Current: 0.5%Ib
Insulation Performance: AC voltage 2KV for 1 minute, Impulse voltage 6kv

Mechanism of electromechanical induction meter

1 - Voltage coil - many turns of fine wire encased in plastic, connected in parallel with load.
2 - Current coil - three turns of thick wire, connected in series with load.
3 - Stator - concentrates and confines magnetic field.
4 - Aluminum rotor disc.
5 - Rotor brake magnets.
6 - Spindle with worm gear.
7 - Display dials - note that the 1/10, 10 and 1000 dials rotate clockwise while the 1, 100 and 10000 dials rotate counter-clockwise.

Electricity meters operate by continuously measuring the instantaneous voltage (volts) and current (amperes) and finding the product of these to give instantaneous electrical power (watts) which is then integrated against time to give energy used (joules, kilowatt-hours etc.). Meters for smaller services (such as small residential customers) can be connected directly in-line between source and customer. For larger loads, more than about 200 ampere of load, current transformers are used, so that the meter can be located other than in line with the service conductors. The meters fall into two basic categories, electromechanical and electronic.

3.9 1N4007

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**

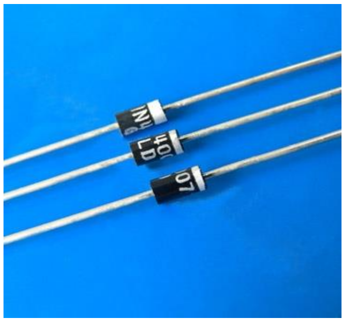


FIG 3.12: 1N4007 diodes

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

- **Diodes of number 1N4001, 1N4002, 1N4003, 1N4004, 1N4005, 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.**

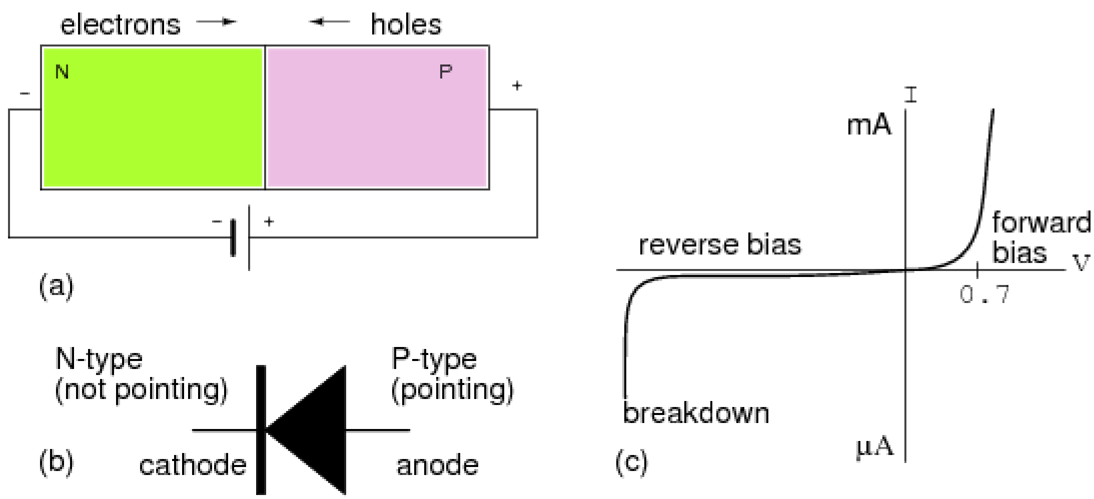


Fig 3.13 :PN Junction diode

3.9.1 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.

3.9.2 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.

3.9.3 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

3.10 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:

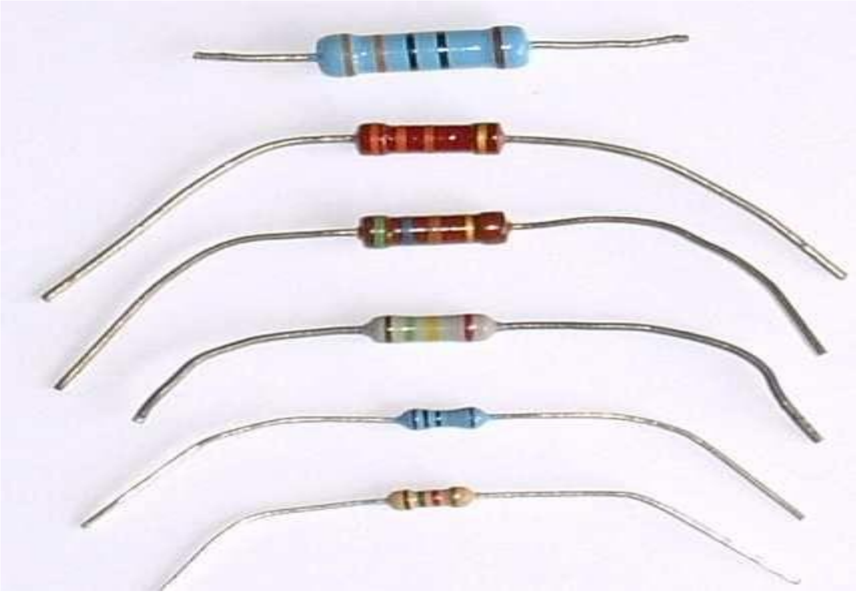
$$V = IR$$

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.

FIG 3.14
Resistors



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:

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

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.

The electrical functionality of a resistor is specified by its resistance: common commercial resistors are manufactured over a range of more than 9 orders of magnitude.

When specifying that resistance in an electronic design, the required precision of the resistance may require attention to the manufacturing tolerance of the chosen resistor, according to its specific application. The temperature coefficient of the resistance may also be of concern in some precision applications. Practical resistors are also specified as having a maximum power rating which must exceed the anticipated power dissipation of that resistor in a particular circuit: this is mainly of concern in power electronics applications. Resistors with higher power ratings are physically larger and may require heat sinking. In a high voltage circuit, attention must sometimes be paid to the rated maximum working voltage of the resistor.

The series inductance of a practical resistor causes its behavior to depart from ohms law; this specification can be important in some high-frequency applications for smaller values of resistance. In a low-noise amplifier or pre-amp the noise characteristics of a resistor may be an issue. The unwanted inductance, excess noise, and temperature coefficient are mainly dependent on the technology used in manufacturing the resistor. They are not normally specified individually for a particular family of resistors manufactured using a particular technology. A family of discrete resistors is also characterized according to its form factor, that is, the size of the device and position of its leads (or terminals) which is relevant in the practical manufacturing of circuits using them.

Units

The ohm (symbol: Ω) 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 mΩ = 10^{-3} Ω), kilohm (1 kΩ = 10^{3} Ω), and megohm (1 MΩ = 10^{6} Ω) 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: $S = \Omega^{-1}$. 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.

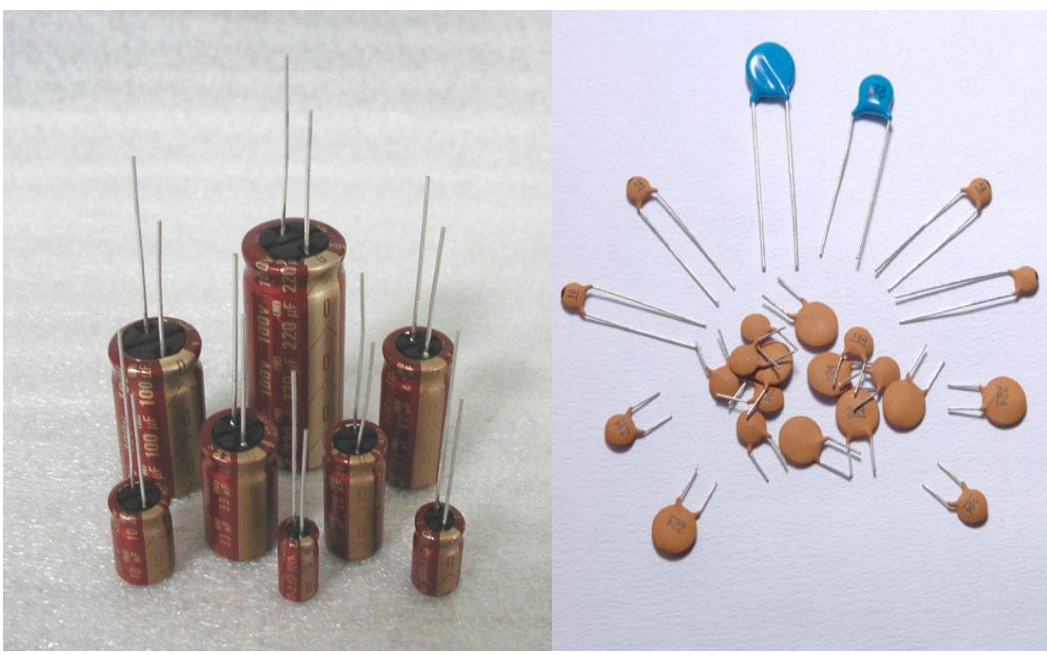
3.11 CAPACITORS

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.

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.

Fig 3.15 Capacitors



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.

Theory of operation

Capacitance

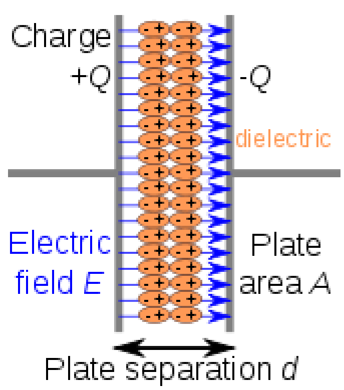


FIG 3.16 Charge separation in a parallel-plate capacitor causes an internal electric field. A dielectric (orange) reduces the field and increases the capacitance.

FIG 3.17 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:

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

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

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

Energy storage

Work must be done by an external influence to "move" charge between the conductors in a capacitor. When the external influence is removed the charge separation persists in the electric field and energy is stored to be released when the charge is allowed to return to its

equilibrium position. The work done in establishing the electric field, and hence the amount of energy stored, is given by:

$$W = \int_{q=0}^{Q} V\,\mathrm{d}q = \int_{q=0}^{Q} \frac{q}{C}\,\mathrm{d}q = \frac{1}{2}\frac{Q^2}{C} = \frac{1}{2}CV^2 = \frac{1}{2}VQ.$$

Current-voltage relation

The current i(t) through any component in an electric circuit is defined as the rate of flow of a charge q(t) passing through it, but actual charges, electrons, cannot pass through the dielectric layer of a capacitor, rather an electron accumulates on the negative plate for each one that leaves the positive plate, resulting in an electron depletion and consequent positive charge on one electrode that is equal and opposite to the accumulated negative charge on the other. Thus the charge on the electrodes is equal to the integral of the current as well as proportional to the voltage as discussed above. As with any anti derivative, a constant of integration is added to represent the initial voltage v (t_0). This is the integral form of the capacitor equation,

$$v(t) = \frac{q(t)}{C} = \frac{1}{C}\int_{t_0}^{t} i(\tau)\,\mathrm{d}\tau + v(t_0).$$

Taking the derivative of this, and multiplying by C, yields the derivative form,

$$i(t) = \frac{\mathrm{d}q(t)}{\mathrm{d}t} = C\frac{\mathrm{d}v(t)}{\mathrm{d}t}.$$

The dual of the capacitor is the inductor, which stores energy in the magnetic field rather than the electric field. Its current-voltage relation is obtained by exchanging current and voltage in the capacitor equations and replacing C with the inductance L.

3.12 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.

3.12.1 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 microcontroller 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 $50
- 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, Diecimila, and 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.**

Arduino Mega 2560

The Mega is the second most commonly encountered version of the Arduino family. The Arduino Mega is like the Arduino Uno's beefier older brother. It boasts 256 KB of memory (8 times more than the Uno). It also had 54 input and output pins, 16 of which are analog pins, and 14 of which can do PWM. However, all of the added functionality comes at the cost of a slightly larger circuit board. It may make your project more powerful, but it will also make your project larger. Check out the official Arduino Mega 2560 page for more details.

Arduino Mega ADK

This specialized version of the Arduino is basically an Arduino Mega that has been specifically designed for interfacing with Android smartphones.

Arduino LilyPad

The LilyPad was designed for wearable and e-textile applications. It is intended to be sewn to fabric and connected to other sewable components using conductive thread. This board requires the use of a special FTDI-USB TTL serial programming cable. For more information, the Arduino LilyPad page is a decent starting point.

Some people think of the entire Arduino board as a microcontroller, but this is inaccurate. The Arduino board actually is a specially designed circuit board for programming and prototyping with Atmel microcontrollers.

The nice thing about the Arduino board is that it is relatively cheap, plugs straight into a computer's USB port, and it is dead-simple to setup and use (compared to other development boards).

Some of the key features of the Arduino Uno include:

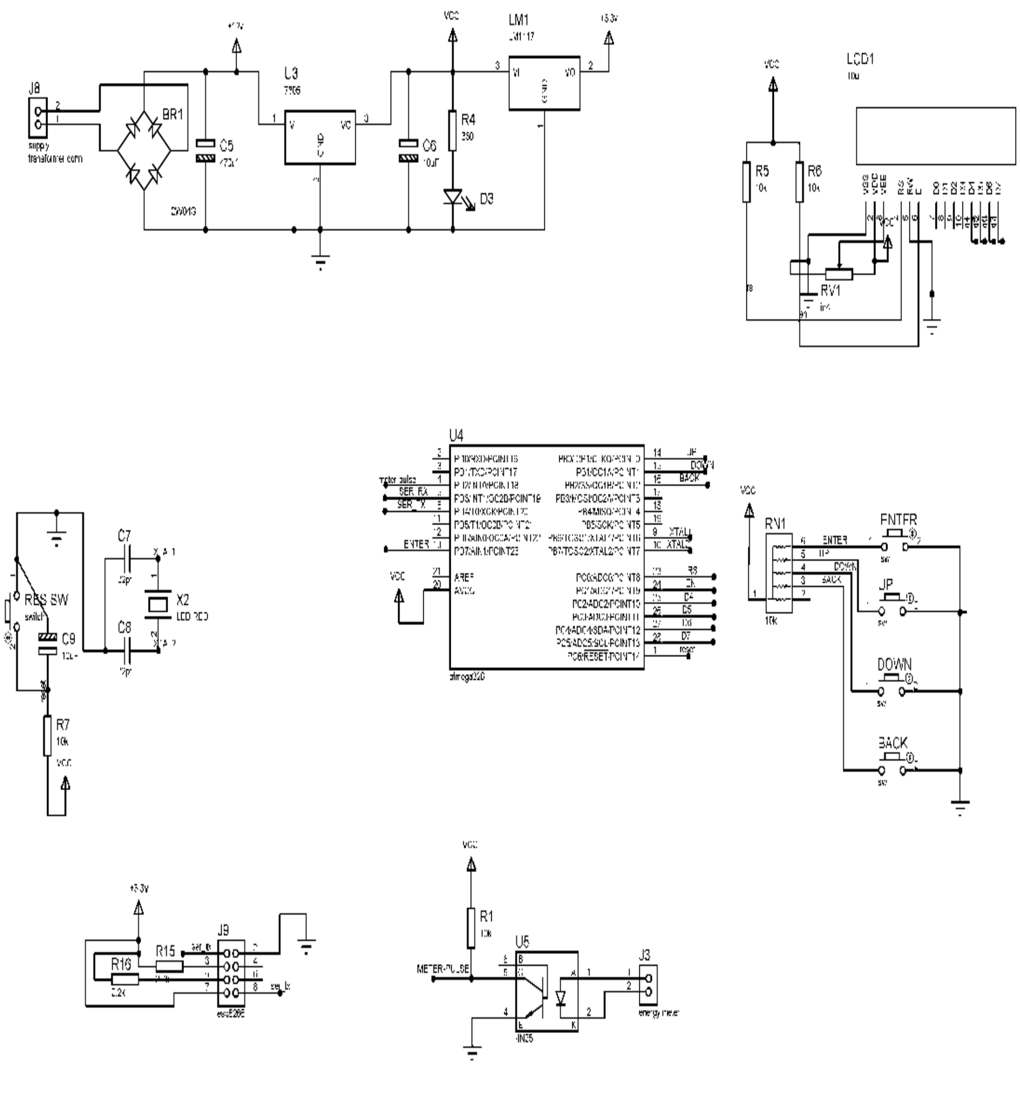
- **An open source design. The advantage of it being open source is that it has a large community of people using and troubleshooting it. This makes it easy to find someone to help you debug your projects.**
- **An easy USB interface . The chip on the board plugs straight into your USB port and registers on your computer as a virtual serial port. This allows you to interface with it as through it were a serial device. The benefit of this setup is that serial communication is an extremely easy (and time-tested) protocol, and USB makes connecting it to modern computers really convenient.**
- **Very convenient power management and built-in voltage regulation. You can connect an external power source of up to 12v and it will regulate it to both 5v and 3.3v. It also can be powered directly off of a USB port without any external power.**
- **An easy-to-find, and dirt cheap, microcontroller "brain." The ATmega328 chip retails for about $2.88 on Digikey. It has countless number of nice hardware features like timers, PWM pins, external and internal interrupts, and multiple sleep modes. Check out the official datasheet for more details.**
- **A 16mhz clock. This makes it not the speediest microcontroller around, but fast enough for most applications.**
- **32 KB of flash memory for storing your code.**
- **13 digital pins and 6 analog pins. These pins allow you to connect external hardware to your Arduino. These pins are key for extending the computing capability of the Arduino into the real world. Simply plug your devices and sensors into the sockets that correspond to each of these pins and you are good to go.**
- **An ICSP connector for bypassing the USB port and interfacing the Arduino directly as a serial device. This port is necessary to re-bootload your chip if it corrupts and can no longer talk to your computer.**

- **An on-board LED attached to digital pin 13 for fast an easy debugging of code.**
- **And last, but not least, a button to reset the program on the chip.**

CHAPTER 4

FABRICATION & TESTING

4.1 SCHEMATIC DIAGRAM



4.2 DESCRIPTION

POWER SUPPLY

The circuit uses standard power supply comprising of a step-down transformer from 230Vto 12V and 4 diodes forming a bridge rectifier that delivers pulsating dc which is then filtered by an electrolytic capacitor of about 470μF to 1000μF. The filtered dc being unregulated, IC LM7805 is used to get 5 V DC constant at its pin no 3 irrespective of input DC varying from 7 V to 15 V. The input dc shall be varying in the event of input ac at 230volts section varies from 160 V to 270 V in the ratio of the transformer primary voltage V1 to secondary voltage V2 governed by the formula V1/V2=N1/N2. As N1/N2 i.e. no. of turns in the primary to the no. of turns in the secondary remains unchanged V2 is directly proportional to V1.Thus if the transformer delivers 12V at 220V input it will give 8.72V at 160V.Similarly at 270 V it will give 14.72 V. Thus the dc voltage at the input of the regulator changes from about 8V to 15V because of A.C voltage variation from 160V to 270V the regulator output will remain constant at 5V.

The regulated 5V DC is further filtered by a small electrolytic capacitor of 10μF for any noise so generated by the circuit. One LED is connected of this 5V point in series with a current limiting resistor of 330Ω to the ground i.e., negative voltage to indicate 5V power supply availability. The unregulated 12V point is used for other applications as and when required.

4.3 STANDARD CONNECTIONS TO 8051 SERIES MICRO CONTROLLER

ATMEL series of 8051 family of micro controllers need certain standard connections. The actual number of the Microcontroller could be “89C51” , “89C52”, “89S51”, “89S52”, and as regards to 20 pin configuration a number of “89C2051”. The 4 set of I/O ports are used based on the project requirement. Every microcontroller requires a timing reference for its internal program execution therefore an oscillator needs to be functional with a desired frequency to obtain the timing reference as t =1/f.

A crystal ranging from 2 to 20 MHz is required to be used at its pin number 18 and 19 for the internal oscillator. It may be noted here the crystal is not to be understood as crystal oscillator It is just a crystal, while connected to the appropriate pin of the microcontroller it results in oscillator function inside the microcontroller. Typically 11.0592 MHz crystal is used in general for most of the circuits using 8051 series microcontroller. Two small value ceramic capacitors of 33pF each is used as a standard connection for the crystal as shown in the circuit diagram.

RESET

Pin no 9 is provided with anre-set arrangement by a combination of an electrolytic capacitor and a register forming RC time constant. At the time of switch on, the capacitor gets charged, and it behaves as a full short circuit from the positive to the pin number 9. After the capacitor gets fully charged the current stops flowing and pin number 9 goes low which is pulled down by a 10k resistor to the ground. This arrangement of reset at pin 9 going high initially and then to logic 0 i.e., low helps the program execution to start from the beginning. In absence of this the program execution could have taken place arbitrarily anywhere from the program cycle. A pushbutton switch is connected across the capacitor so that at any given time as desired it can be pressed such that it discharges the capacitor and while released the capacitor starts charging again and then pin number 9 goes to high and then back to low, to enable the program execution from the beginning. This operation of high to low of the reset pin takes place in fraction of a second as decided by the time constant R and C.

For example: A 10µF capacitor and a 10kΩ resistor would render a 100ms time to pin number 9 from logic high to low, there after the pin number 9 remains low.

External Access(EA):

Pin no 31 of 40 pin 8051 microcontroller termed as EA¯ is required to be connected to 5V for accessing the program form the on-chip program memory. If it is connected to ground then the controller accesses the program from external memory. However as we are using the internal memory it is always connected to +5V.

4.3.1 OPTOCOUPLER

Optocoupler is a 6 pin IC. It is a combination of 1 LED and a transistor. Pin 6 of transistor is not generally used and when light falls on the Base-Emitter junction then it switches and pin5 goes to zero.

If input of the diode is zero and other end of diode is GND then the output is one.

- **When logic zero is given as input then the light doesn't fall on transistor so it doesn't conduct which gives logic zero as output.**
- **When logic 1 is given as input then light falls on transistor so that it conducts, that makes transistor switched ON and it forms short circuit this makes the output is logic zero as collector of transistor is connected to ground.**

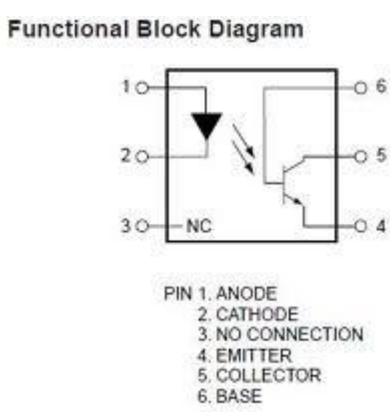


Fig 4.2

MAX232

The MAX232 used in the project is an integrated circuit that converts signals from an RS-232 serial port to signals suitable for use in TTL compatible digital logic circuits like microcontroller. The MAX232 is a dual driver/receiver and typically converts the RX, TX, CTS and RTS signals

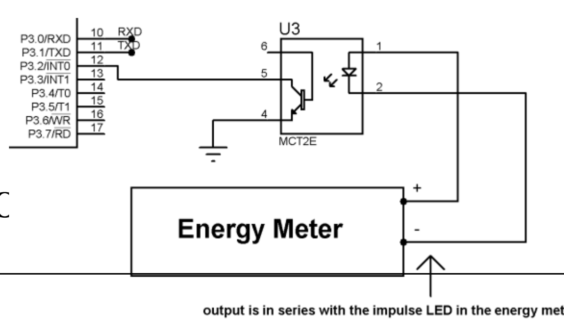
4.3.2 OPERATION

Connections:

The output of power supply which is 5v is connected to 40th pin of microcontroller and Gnd is connected to 20th pin of microcontroller. Pin 2.0 to pin 2.7 of port 2 of MC is connected to data pins i.e., D_0 to D_7 of LCD display. Pins 4, 5, 6 i.e., RS, Rw, EN. Of LCD are given to p 0.0 to p 0.2 of port of 0 of MC. Pin 3.0 and pin 3.1 of port 3 of MC are connected to pin 11 and pin 12 of Max232. Pin 3.3 of port 3 of MC is connected to pin 5 of Opto coupler. Pin 1 & 2 of Opto coupler are connected to energy meter. Pin 14 & 13 of Max232 are given to pins 2 & 3 of DB9 male connector. Pin 2 & pin 5 of DB9 Female Connector are given to GSM modem.

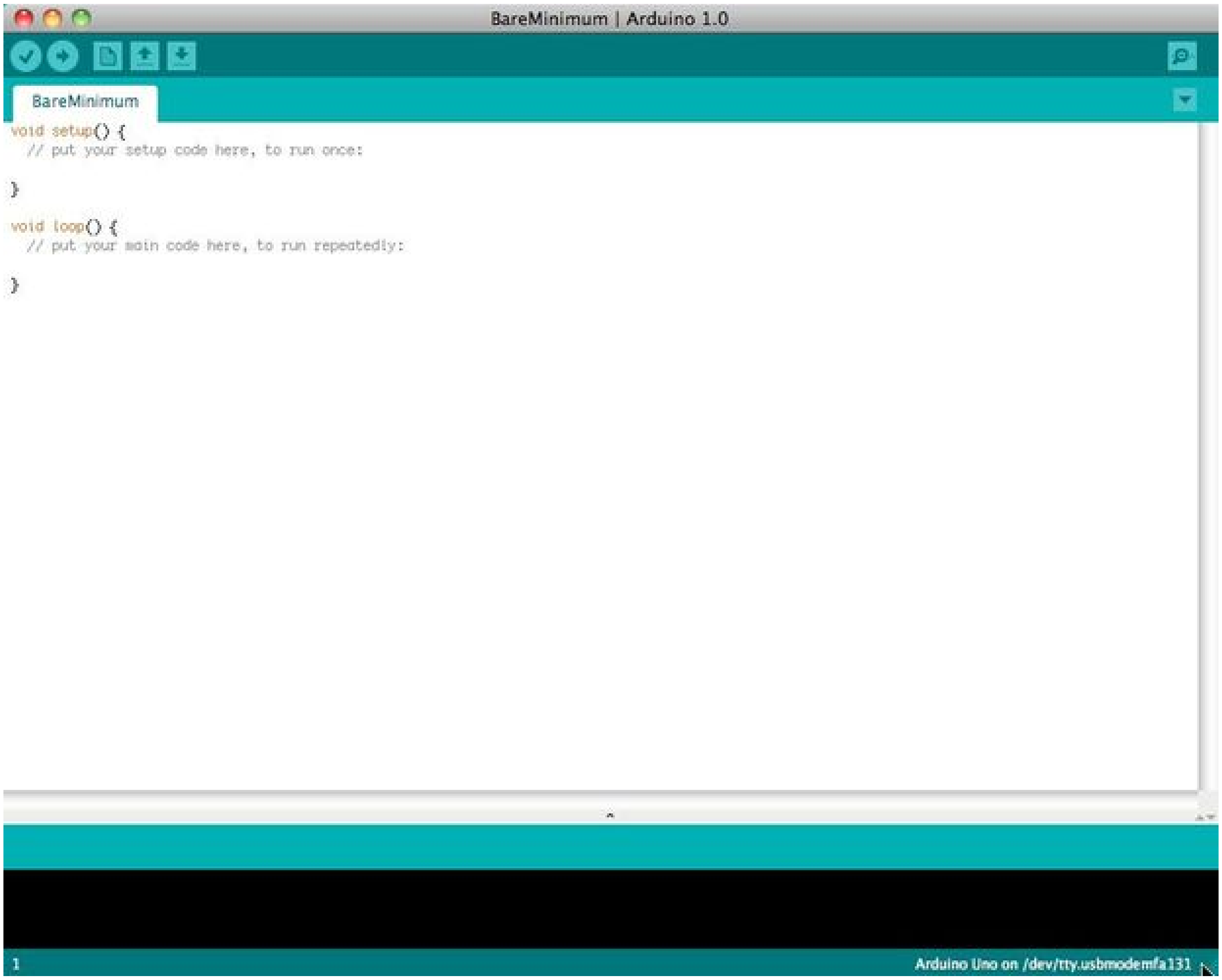
Working:

FIG 4.3



The project uses a commercial digital energy meter and derives positive pulses from the same by
Using an opto isolator led in series with the energy pulsing LED of the energy meter. The pulsing LED pulses 3200 times for 1 unit of electrical energy i.e., 1 kilo watt hour. Thus the opto led which is in series with the energy pulsating led blinks at the same rate developing logic zero and high pulses across the opto transistor which is interfaced to microcontroller. As it is not feasible to wait for consuming 1000 Wt Hr. the program assumes 10 pulses for unit, which is fed to the microcontroller pin no 12 to send 1 unit consumption through the GSM modem duly interfaced to the microcontroller through Max232, Therefore the 1 unit so sent shown have to be read as 1/3200 unit.Upon a missed a call to the project board GSM modem ,the caller's number gets stored in the microcontroller for further communication to that number only. This gives the unique flexibility for changing number by the user at will without going through the cumbersome process of writing the number while burning the program on to the microcontroller .Thus in that case only that number is used for communication and the user has no option to change that .

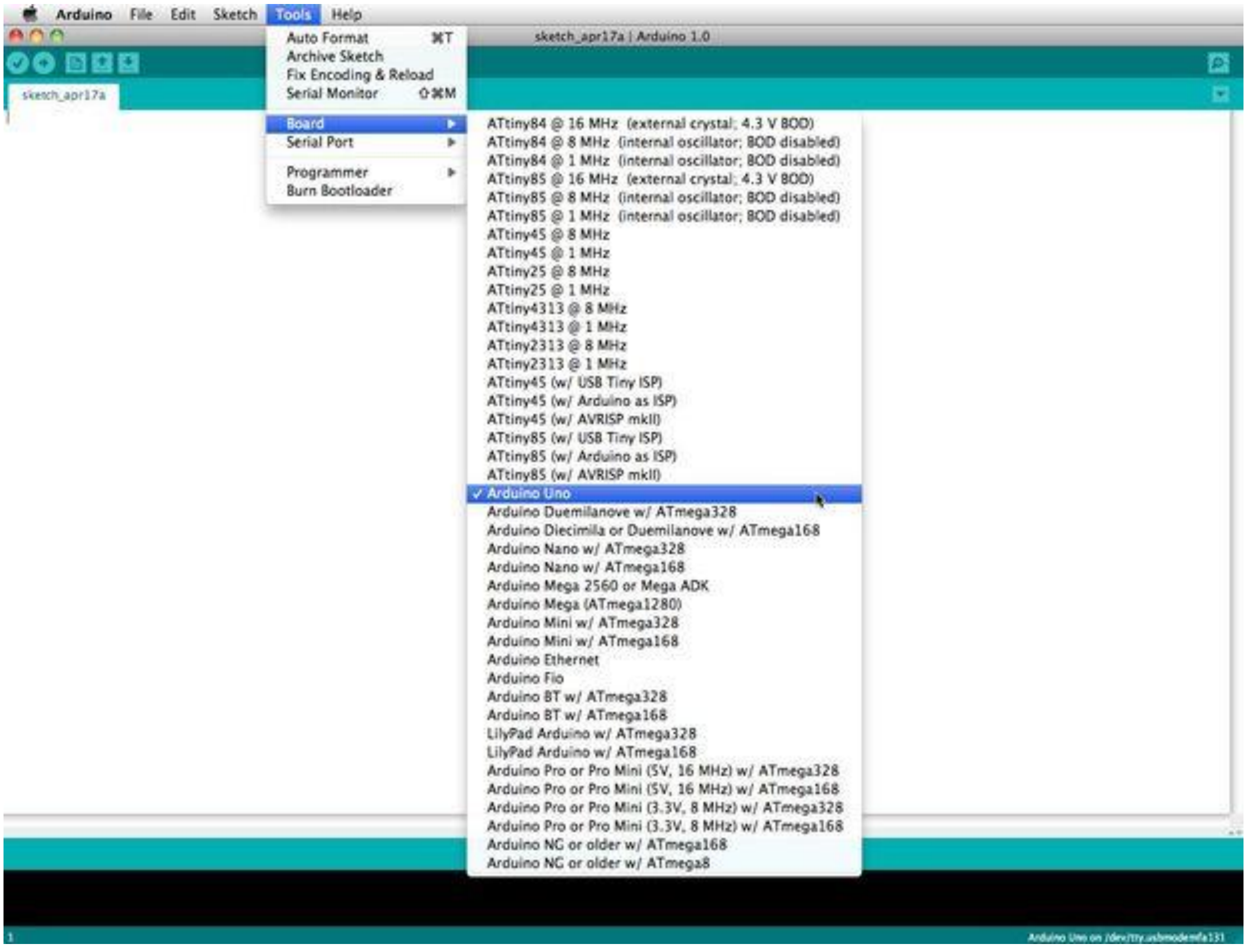
4.4 Arduino IDE

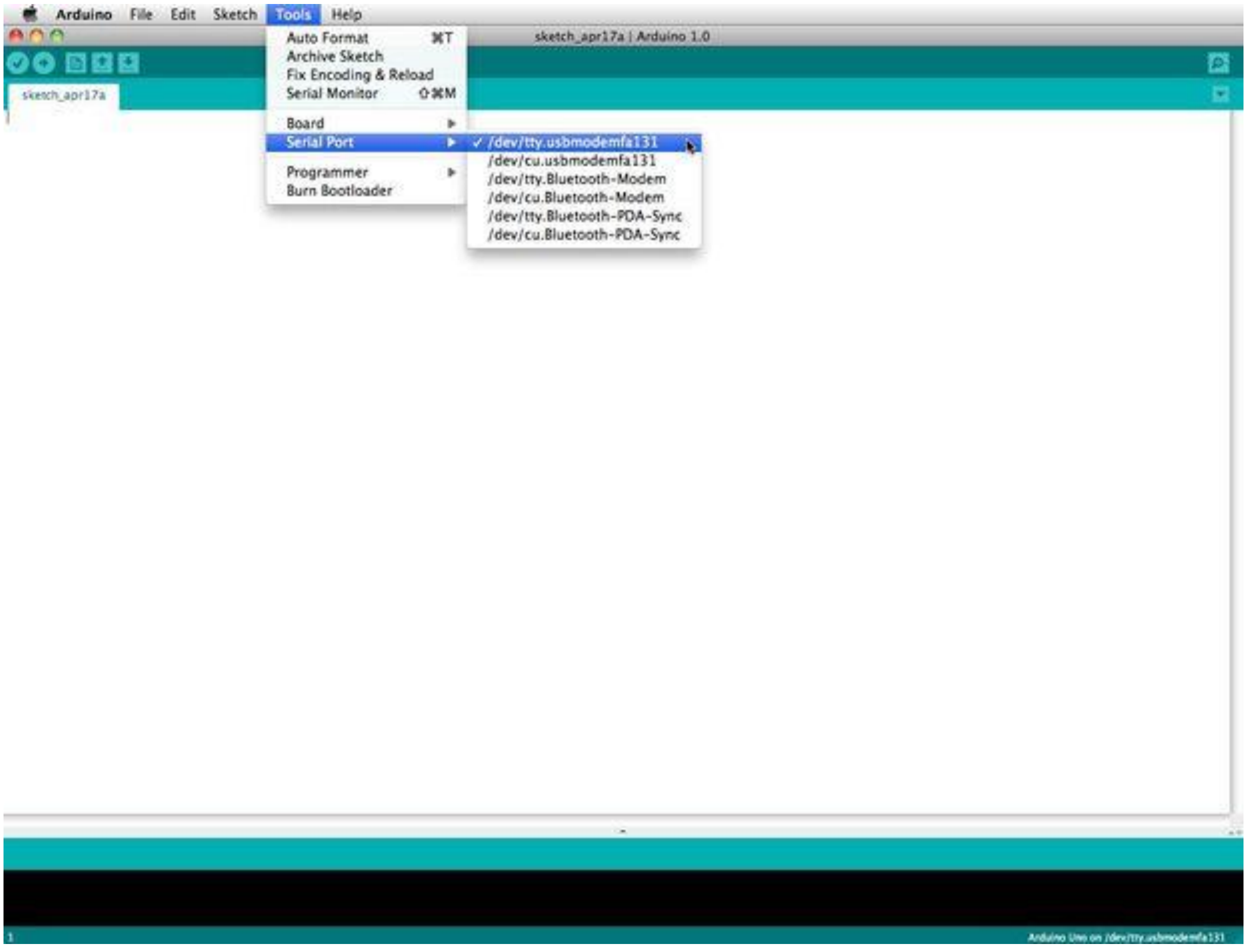


Before you can start doing anything with the Arduino, you need to download and install the Arduino IDE (integrated development environment). From this point on we will be referring to the Arduino IDE as the Arduino Programmer.

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

Settings


Arduino File Edit Sketch Tools Help
sketch_apr17a | Arduino 1.0
sketch_apr17a
Auto Format ⌘T
Archive Sketch
Fix Encoding & Reload
Serial Monitor ⇧⌘M
Board
Serial Port
Programmer
Burn Bootloader
ATtiny84 @ 16 MHz (external crystal; 4.3 V BOD)
ATtiny84 @ 8 MHz (internal oscillator; BOD disabled)
ATtiny84 @ 1 MHz (internal oscillator; BOD disabled)
ATtiny85 @ 16 MHz (external crystal; 4.3 V BOD)
ATtiny85 @ 8 MHz (internal oscillator; BOD disabled)
ATtiny85 @ 1 MHz (internal oscillator; BOD disabled)
ATtiny45 @ 8 MHz
ATtiny45 @ 1 MHz
ATtiny25 @ 8 MHz
ATtiny25 @ 1 MHz
ATtiny4313 @ 8 MHz
ATtiny4313 @ 1 MHz
ATtiny2313 @ 8 MHz
ATtiny2313 @ 1 MHz
ATtiny45 (w/ USB Tiny ISP)
ATtiny45 (w/ Arduino as ISP)
ATtiny45 (w/ AVRISP mkII)
ATtiny85 (w/ USB Tiny ISP)
ATtiny85 (w/ Arduino as ISP)
ATtiny85 (w/ AVRISP mkII)
✓ Arduino Uno
Arduino Duemilanove w/ ATmega328
Arduino Diecimila or Duemilanove w/ ATmega168
Arduino Nano w/ ATmega328
Arduino Nano w/ ATmega168
Arduino Mega 2560 or Mega ADK
Arduino Mega (ATmega1280)
Arduino Mini w/ ATmega328
Arduino Mini w/ ATmega168
Arduino Ethernet
Arduino Fio
Arduino BT w/ ATmega328
Arduino BT w/ ATmega168
LilyPad Arduino w/ ATmega328
LilyPad Arduino w/ ATmega168
Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328
Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168
Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328
Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168
Arduino NG or older w/ ATmega168
Arduino NG or older w/ ATmega8



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

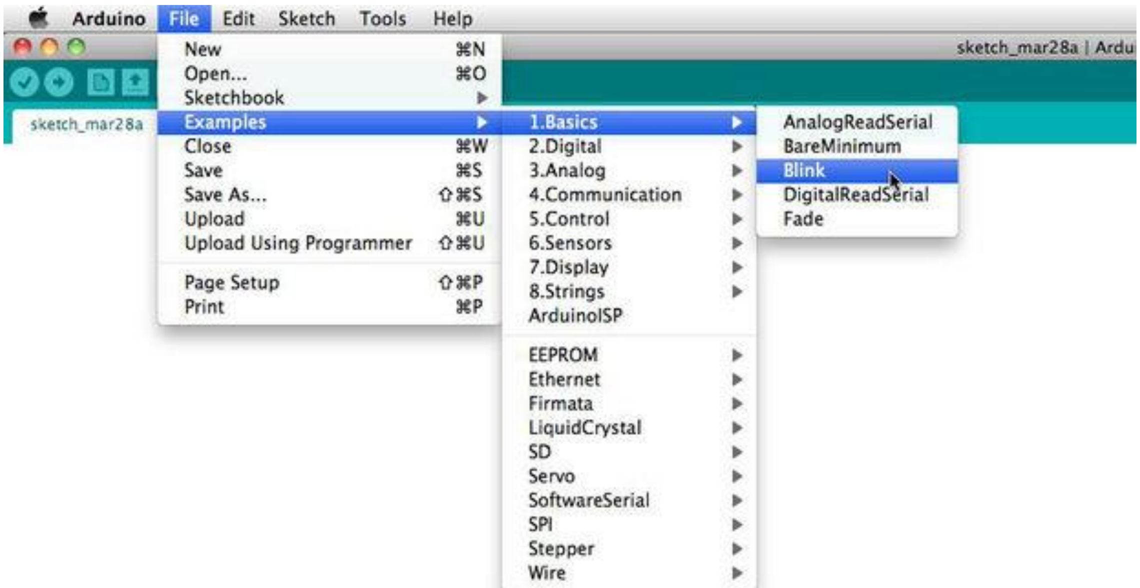
Select the version of board that you are using. Since I have an Arduino Uno plugged in, I obviously selected "Arduino Uno."

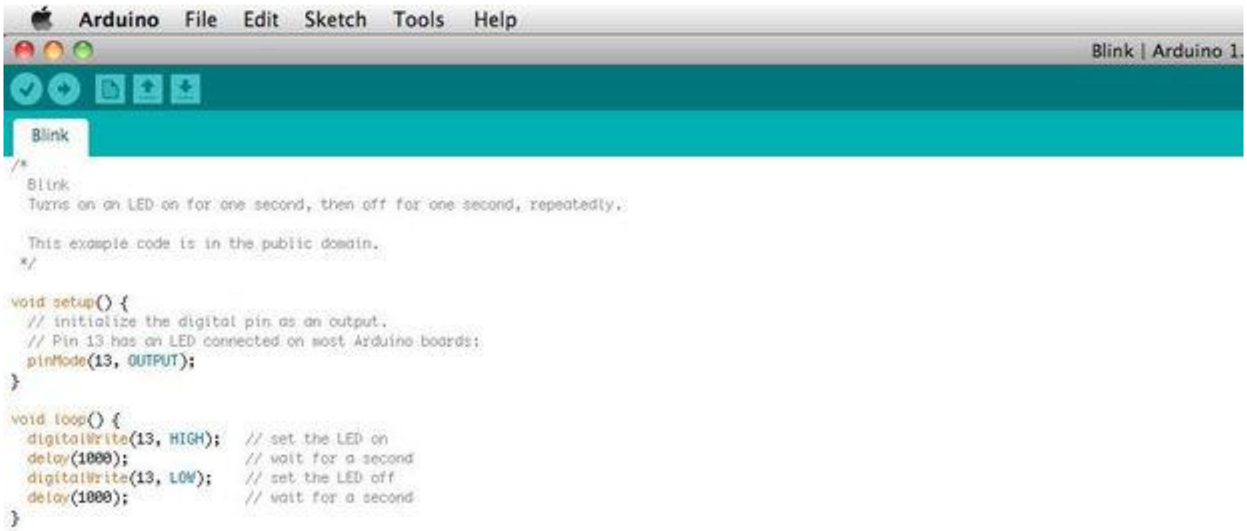
To set the serial port, go to the following:

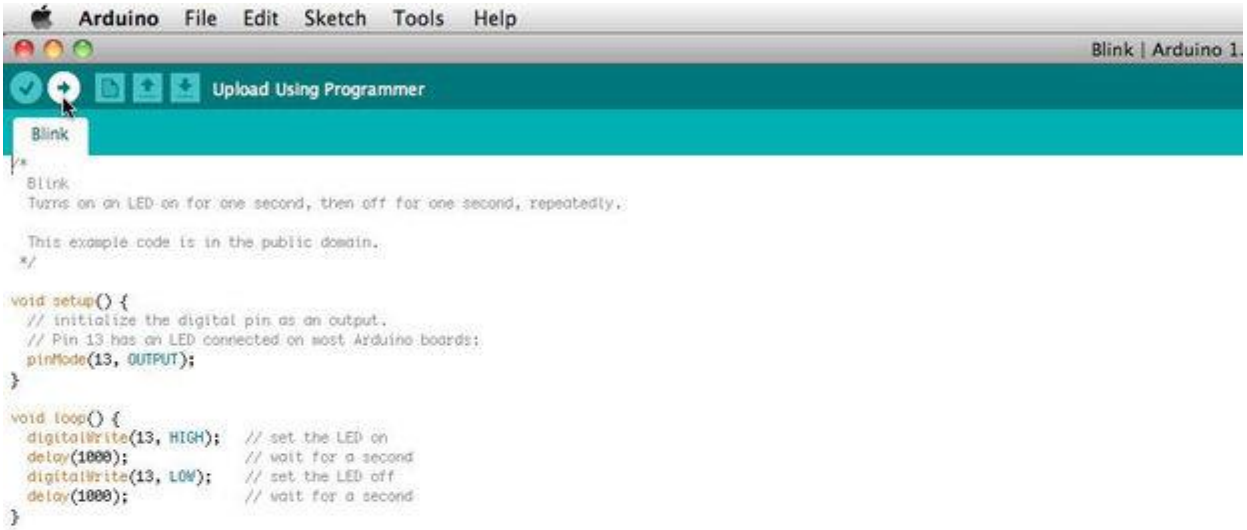
Tools --> Serial Port

Select the serial port that looks like:

/dev/tty.usbmodem [random numbers]


Arduino
File
Edit
Sketch
Tools
Help
sketch_mar28a | Ardu
sketch_mar28a
New ⌘N
Open... ⌘O
Sketchbook
Examples
Close ⌘W
Save ⌘S
Save As... ⇧⌘S
Upload ⌘U
Upload Using Programmer ⇧⌘U
Page Setup ⇧⌘P
Print ⌘P
1.Basics
2.Digital
3.Analog
4.Communication
5.Control
6.Sensors
7.Display
8.Strings
ArduinoISP
EEPROM
Ethernet
Firmata
LiquidCrystal
SD
Servo
SoftwareSerial
SPI
Stepper
Wire
AnalogReadSerial
BareMinimum
Blink
DigitalReadSerial
Fade


Arduino File Edit Sketch Tools Help
Blink | Arduino 1.
Blink
/*
Blink
Turns on an LED on for one second, then off for one second, repeatedly.
This example code is in the public domain.
*/
void setup() {
// initialize the digital pin as an output.
// Pin 13 has an LED connected on most Arduino boards:
pinMode(13, OUTPUT);
}
void loop() {
digitalWrite(13, HIGH); // set the LED on
delay(1000); // wait for a second
digitalWrite(13, LOW); // set the LED off
delay(1000); // wait for a second
}



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:

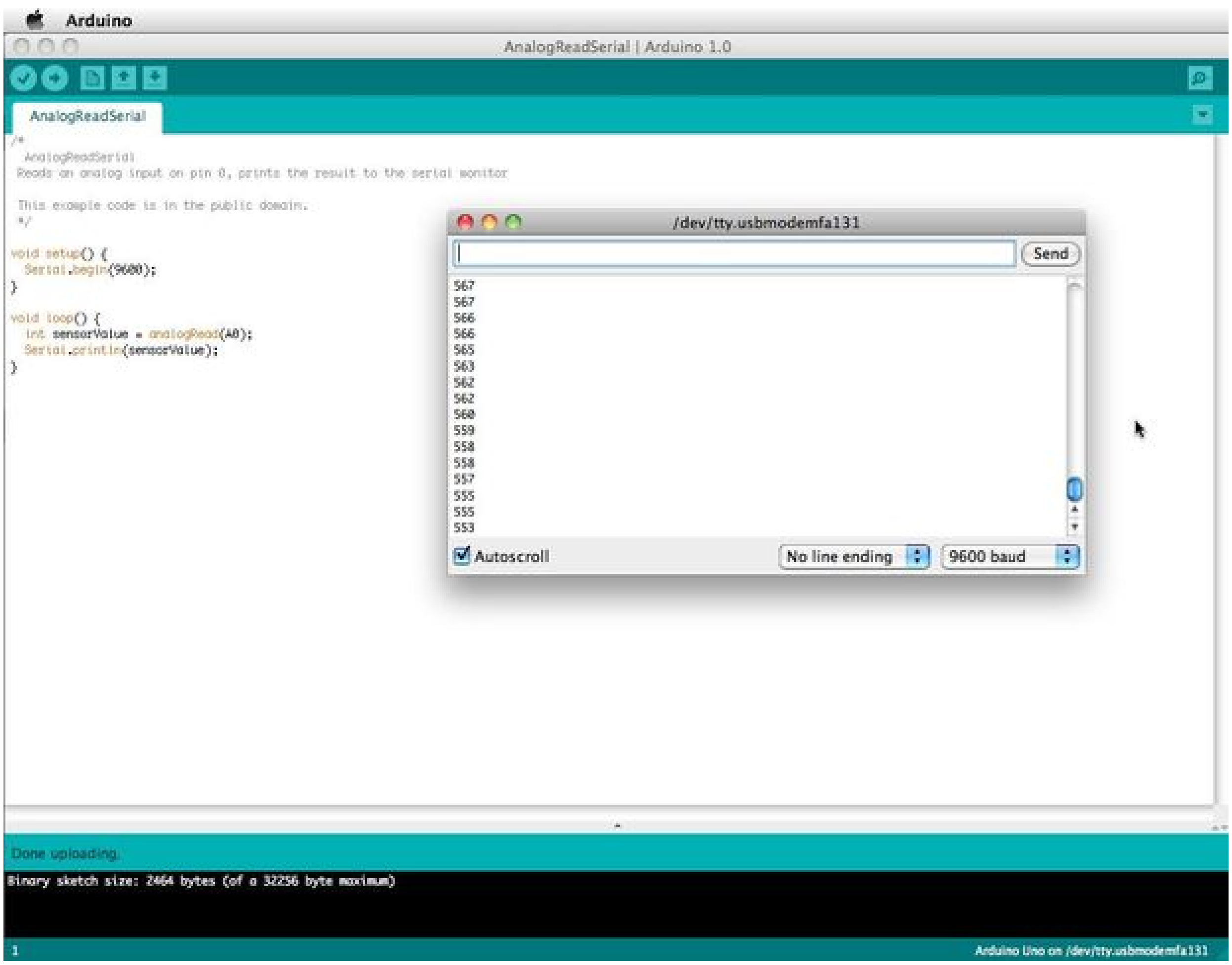
Files --> Examples --> Basics --> Blink

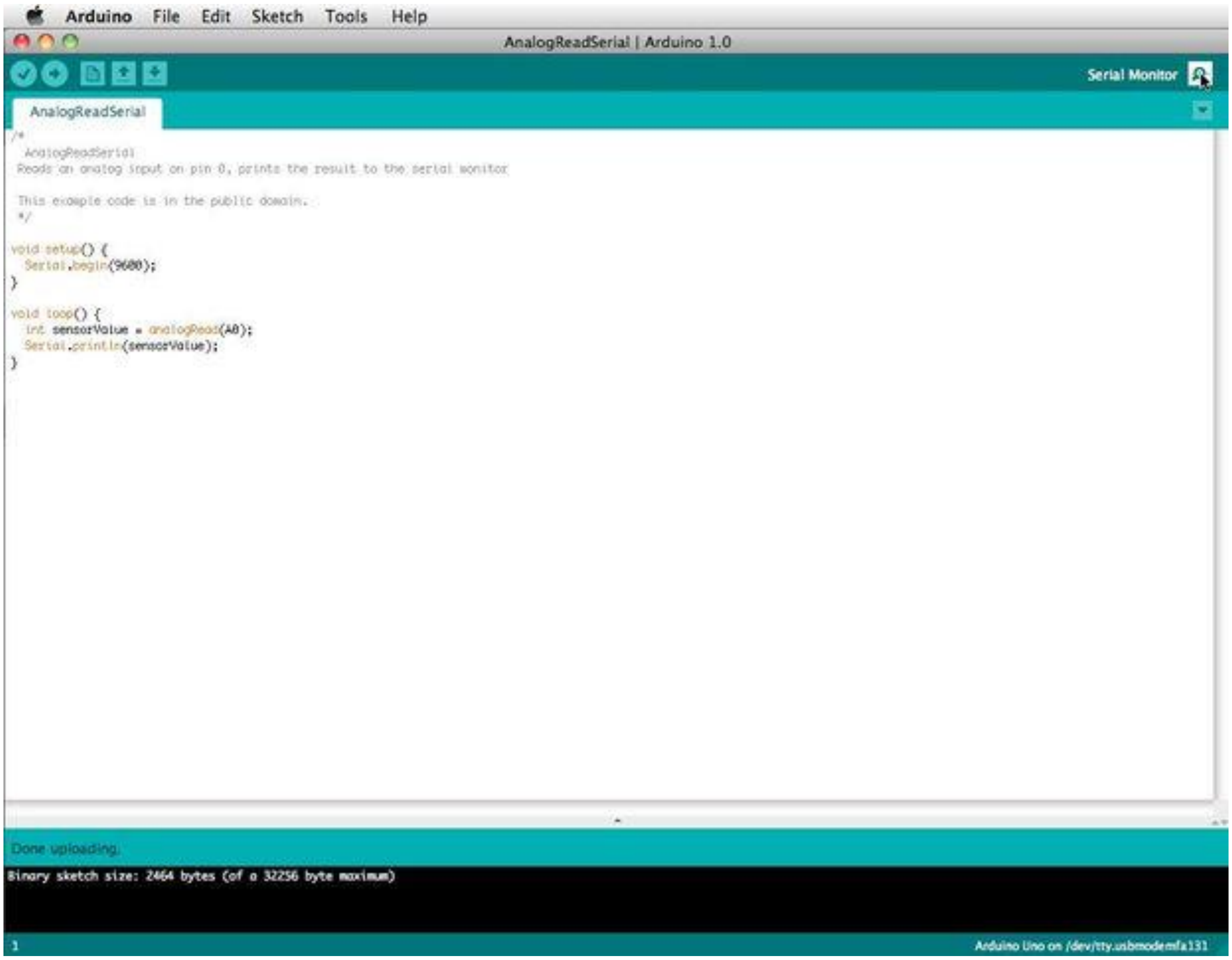
The blink example basically sets pin D13 as an output and then blinks the test LED on the Arduino board on and off every second.

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





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 --> Analog Read Serial

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 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:**

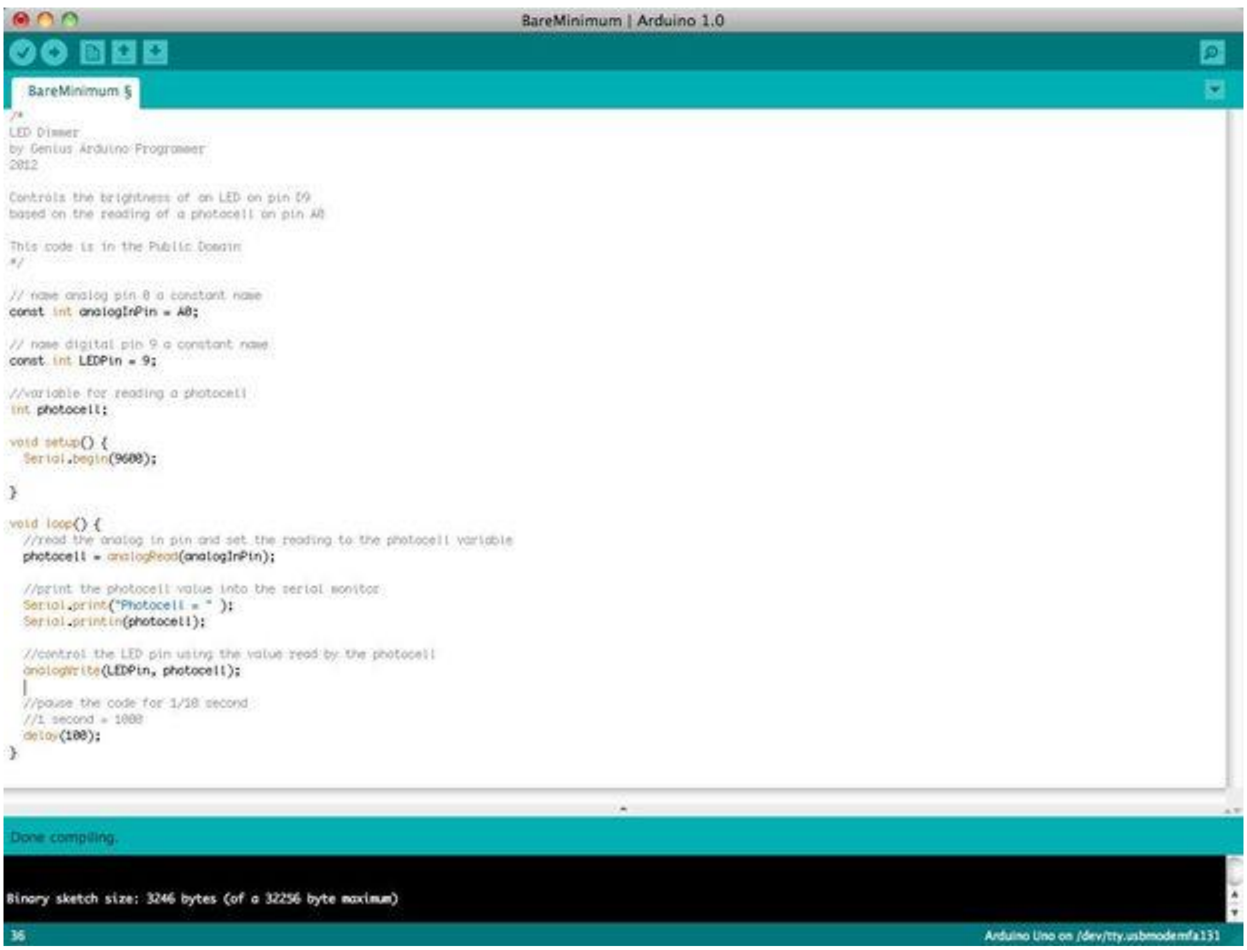
File --> Examples --> 2.Digital --> Button

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.



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 out this 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

The Bare Minimum 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);
}
```

4.5 HARDWARE TESTING

4.5.1 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.

4.5.2 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 5

RESULT & DISCUSSION

This project main aim to detect theft while taking power supply through third party. Atmega 328 plays a key role to upload the data of sensors to the cloud which helps to intimate to customers through blynk app or mail, which can easily operates and takes care through mobile from anywhere.

APPLICATIONS:

Accessing information is easy for customer from cloud through IoT.

Theft detection at consumer end in existent time.

Conclusion

This system helps in control the energy consumption and avoiding energy wastage is very important. This is an Atmega 328 based design and implementation of energy meter by using IOT concept. In the proposed system, current sensor and voltage from mains and load reading system is designed to monitor continuously the meter reading and transfer the reading to certain server. This data can be access from anywhere on the globe at any time.

Future Scope

The project mainly aims at providing overall infrastructure of the energy meter presently used for the smart city concept. The main improvement for the future is going to make energy meter readings, tampering identification techniques, and connection and disconnection and also the pre information providing to the users all is going to happen on Wi-Fi internet.

Where we are going to develop some Wi-Fi hotspots in each area through which all the energy meters are get connected and set 4 to 5 parameters which is also going to be monitored. Also, in future we can go with some standard apps or standard tools, and connect and disconnect of every meter on the on – payment and non – payment that will be fast as compared to the present method.

References

TEXT BOOKS REFERRED

1. ATMEGA 328 Data Sheets.

WEBSITES

- **www.atmel.com**
- **www.beyondlogic.org**
- **www.wikipedia.org**
- **www.howstuffworks.com**
- **www.alldatasheets.com**

Appendix A

REAL TIME CLOCK

A real-time clock (RTC) is an electronic device (most often in the form of an integrated circuit) that measures the passage of time.

Although the term often refers to the devices in personal computers, servers and embedded systems, RTCs are present in almost any electronic device which needs to keep accurate time of day.

Terminology

The term *real-time clock* is used to avoid confusion with ordinary hardware clocks which are only signals that govern digital electronics, and do not count time in human units. RTC should not be confused with real-time computing, which shares its three-letter acronym but does not directly relate to time of day.

Purpose

Although keeping time can be done without an RTC, using one has benefits:

- **Low power consumption (important when running from alternate power)**
- **Frees the main system for time-critical tasks**
- **Sometimes more accurate than other methods**

A GPS receiver can shorten its startup time by comparing the current time, according to its RTC, with the time at which it last had a valid signal. If it has been less than a few hours, then the previous ephemeris is still usable.

Some motherboards are made without real time clocks. The real time clock is omitted either out of the desire to save money (as in the Raspberry Pi system architecture) or because real time clocks may not be needed at all (as in the Arduino system architecture

Power source

RTCs often have an alternate source of power, so they can continue to keep time while the primary source of power is off or unavailable. This alternate source of power is normally a lithium battery in older systems, but some newer systems use a super capacitor, because they are rechargeable and can be soldered. The alternate power source can also supply power to battery backed RAM.

Timing

Most RTCs use a crystal oscillator, but some have the option of using the power line frequency. The crystal frequency is usually 32.768 kHz, the same frequency used in quartz clocks and watches. Being exactly 2 cycles per second, it is a convenient rate to use with simple binary counter circuits. The low frequency saves power, while remaining above human hearing range. The quartz tuning fork of these crystals does not change size much from temperature, so temperature does not change its frequency much.

Some RTCs use a micromechanical resonator on the silicon chip of the RTC. This reduces the size and cost of an RTC by reducing its parts count. Micromechanical resonators are much more sensitive to temperature than quartz resonators. So, these compensate for temperature changes using an electronic thermometer and electronic logic.

Typical crystal RTC accuracy specifications are from ±100 to ±20 parts per million (8.6 to 1.7 seconds per day), but temperature-compensated RTC ICs are available accurate to less than 5 parts per million. In practical terms, this is good enough to perform celestial navigation, the classic task of a chronometer. In 2011, chip-scale atomic clocks became available. Although vastly more expensive and power-hungry (120 mW vs. <1 µW), they keep time within 50 parts per trillion (5×10^{-11}).

Examples

Many integrated circuit manufacturers make RTCs, including Epson, Intersil, IDT, Maxim, NXP Semiconductors, Texas Instruments, STMicroelectronics and Ricoh. A common RTC used in single-board computers is the Maxim Integrated DS1307.

The RTC was introduced to PC compatibles by the IBM PC/AT in 1984, which used a Motorola MC146818 RTC. Later, Dallas Semiconductor made compatible RTCs, which were often used in older personal computers, and are easily found on motherboards because of their distinctive black battery cap and silkscreened logo.

In newer computer systems, the RTC is integrated into the southbridge chip.

Some microcontrollers have a real-time clock built in, generally only the ones with many other features and peripherals.