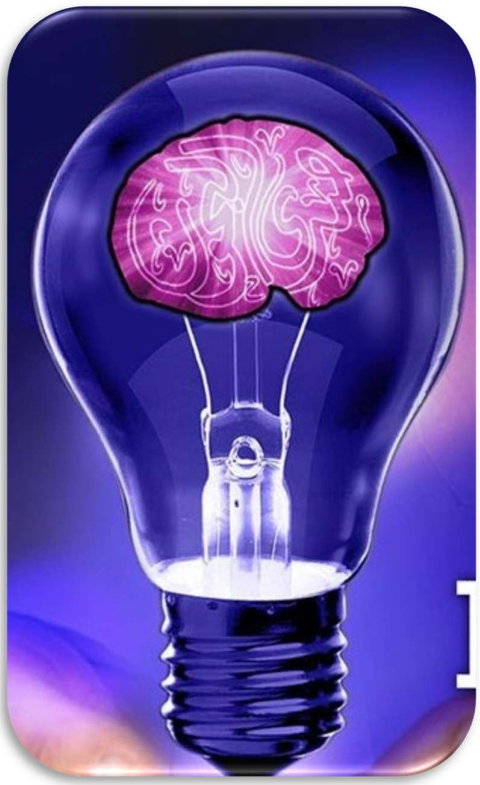


BEES Magazine

February 2017

K S R Institute for Engineering and Technology

Department of Electrical and Electronics Engineering

BEES Magazine

Together We Make Difference

February 2017

Contents

Editorial Board

Robo-Doctor

RANJITHKUMAR S
III year EEE

YOKESHWARAN B
III year EEE

Introduction

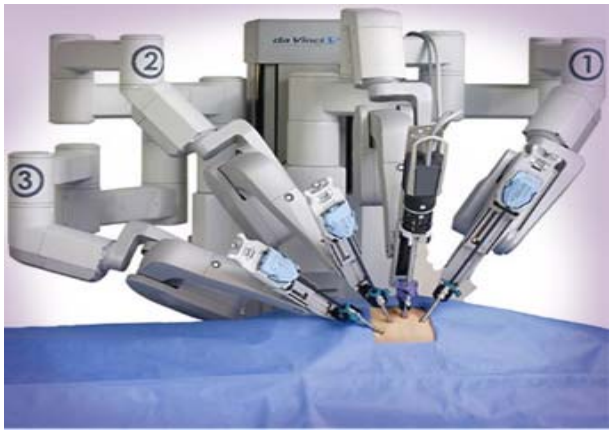
A technological idea born in science fiction is a promising answer to the challenging realities of modern health care. With fewer doctors to meet the growing health-care needs of our aging population, hospitals and health systems are investing in robotic systems for surgery and telemedicine that increase their patient capacity and geographic reach. Machines like the da Vinci Surgical System (Intuitive Surgical Systems, Sunnyvale, CA) and the RP-7i Remote Presence medical robot (InTouch Health, Santa Barbara, CA) connect patients who need specialized care with physicians who can help them – even if they are an ocean apart.

But fiction soon collided with reality when the nuclear industry invented a real gadget, nicknamed a Waldo, for the safe manipulation of radioactive materials from a remote location, and a new industry was born.

Seventy years later, medical robots are still an emerging technology. But forces such as health-care reform, the shortage of doctors and nurses, and the skyrocketing costs of hospital care are driving its acceptance like never before.

Robo-Surgeon

But it's about more than just saving money. Advocates of robotic surgery, for example, claim the da Vinci surgical robot achieves significantly better outcomes than either radiation or traditional surgery in delicate procedures such as radical prostatectomy for prostate cancer. They say robotic surgery can remove more cancerous tissue with less disruption of adjacent nerve endings than other methods, helping to reduce cancer recurrence and retain sexual function. That's why some 85% of men undergoing prostate cancer surgery are choosing medical centres that offer robotic surgery. Introduced in 1999, the da Vinci system remains the standard robotic system for complex operations in cardiac, colorectal, gynecologic, thoracic, urologic, and head and neck surgeries. The U.S. Food & Drug Administration continues to approve its use in additional surgical applications.



The guts of the system include four robotic arms, a high-definition 3-D viewing system with up to 10x magnification, and a novel family of specialized instruments with Intuitive Surgical's proprietary "EndoWrist" technology. Traditional devices such as forceps, scalpels, retractors, and suture drivers have been reimagined for the robotic age, with seven degrees of freedom, a large range of motion, and less risk from surgeon hand tremors.

The system's robotic and computer technologies work together to scale, filter, and translate the surgeon's hand movements into micro-movements that guide the instruments, not unlike the Waldo of science fiction. Seated at a viewing and control console located in or near the operating room, the surgeon uses hand controls to manipulate surgical instruments through tiny incisions. The instruments move like high-precision puppets with each motion of the surgeon's hand, wrist, or finger.

Robotic surgery has its critics, especially among those concerned about its comparatively high cost and the worry that hospitals will over-hype the technology to lure patients and recoup their investments.

Catherine Mohr, director of medical research at Intuitive Surgical, acknowledged that a typical system "will cost you about as much as a solid gold surgeon. It's a fairly big capital investment, but once you've got it, your procedure costs do come down."

The next challenges in robotic surgery are to make the technique faster and easier to use in more complex operations, which is key to their eventual routine, cost-effective use. She said she is working with prototype designs that eliminate the need to move the robot to reach additional areas of the body and add new visualization capabilities

Robotic Hands Across the Water

The ultimate in robotic surgery would be the integration of daVinci-style surgical robots with telemedicine technologies that enable medical

professionals to consult, assist, supervise, or train their counterparts in distant locations.

Intuitive Surgical says the daVinci is theoretically capable of long-distance surgery, but it's not the company's current focus. But in terms of experimentation, remote operations date back more than a decade.

Remote presence robots bring big-city know-how to small-town clinics and trauma centers. The impact can be life-saving in the case of emergencies such as stroke or heart attack, where a fast diagnosis and onset of treatment is critical to saving heart or brain function. Telemedicine increases the public's access to advanced expertise while helping to reduce the overall cost of care. Critical care doctors in major trauma centers can evaluate accident victims remotely and, often, eliminate the need to transport them to larger hospitals.

The RP-7i system features one or more physician control stations linked wirelessly to what the company calls an "endpoint": a remote-controlled mobile console/medical cart topped by a high-definition video screen and camera. The robot enables two-way doctor-to-doctor and doctor-to-patient communication and visualization. It is equipped with a suite of basic medical instruments to allow remote monitoring of vital signs. The consulting doctor can observe patient behavior, check bedside monitors, confer with family members, or review medical images with the patient. Through his or her robotic counterpart, the remote physician can travel from room to room and to the nurses' station to review care plans. With doctors in short supply, especially in rural areas, technologies that help them be in two places at once will surely be part of tomorrow's health-care landscape. As Wang says, "We have to innovate our way out of this problem."

Getting Started with the Raspberry Pi

BHUVANAKUMAR N
IV year EEE

SANJAYKUMAR C
IV year EEE

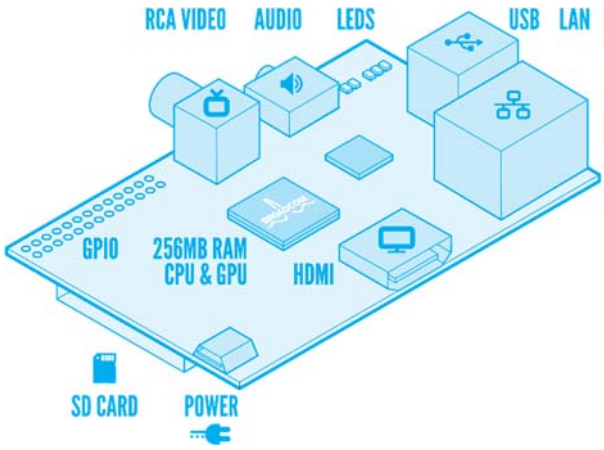
Introduction

The super low cost computer called the Raspberry Pi is mind-blowing and awesome. As TechCrunch recently reported, the minicomputer on a circuit board is designed to give kids around the globe an easy way to learn computer programming. But the Raspberry Pi is not like a computer you get from Apple or pickup at the local Best Buy. It's not as simple as plug and play. It takes a bit of extra work and time. And you need to be a hacker before you can get it working.

The Raspberry Pi Foundation, the British-based not-for-profit charity that makes the computer, says it wants to keep the price down, perhaps even $20 one day. But to keep the price down, they had to make some hard choices about what hardware and interfaces to include.

While a $25 to $35 computer makes a nice headline, the odds are good it's going to cost you several times that amount to get it going. That full amount is still insanely lower than the price of a new PC or Mac.

You will need to connect a power supply, monitor, USB keyboard, mouse, SD memory card, and an Ethernet connection. If you want to get fancy, you might also find a box to hold it and attach all sorts of sensors, displays, and almost anything you can think of. No soldering is required to get the basic system running.



The distributors selling the Raspberry Pi also sell some of the needed accessories, but they are in high demand and short supply, just like the circuit board itself.

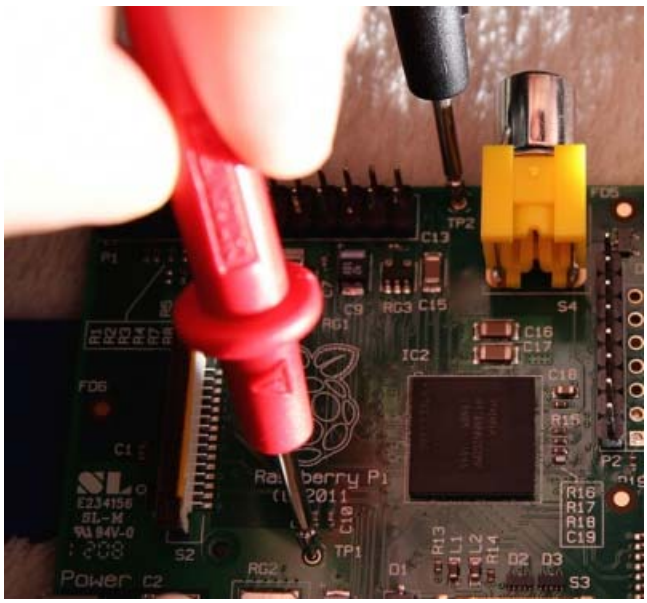
Power

The quick start guide says you need to find a "Micro USB power supply – make sure you use a good quality one, capable of providing at least 700mA at 5V. Do not attempt to power your Raspberry Pi by plugging it into a computer or a hub."

The Raspberry Pi makers say they picked the Micro USB jack for its ubiquity, saying it was chosen by the major mobile operator trade group,

GSMA, as the Universal Charging Solution. Micro USB is also the standard charging port in Europe and it's used on Android phones in the US.

Looking at the fine print on the power adapter, it read "Output: 4.9V and 0.85 A." It wasn't the 5V in the spec but it sounded close enough, and 0.85 A was greater than the 700mA spec. This power supply did get the Raspberry Pi LED's to light up. And it did work for a bit. But after everything was connected,



With the Amazon Kindle power supply, got a reading of 4.62 volts. The wiki said I should see a voltage between 4.75 and 5.25 volts and "anything outside this range indicates that you have a problem with your power supply or your power cable, or the input polyfuse F3."

Monitor

The video output from the Raspberry Pi is either HDMI or composite video via an RCA jack. You can't connect to a monitor using VGA, which many users might first try. The Raspberry Pi FAQ says "there is no VGA support, but adapters are available, although these are relatively expensive." According the the FAQ, the reason there is no VGA support is "the chip specifically supports HDMI. VGA is consider to be an end-of-life technology, so supporting it doesn't fit with our plans at the moment." The Raspberry Pi Foundation is hoping many of the students it's trying to reach may have access to an old TV to use as the display since many of them have composite video input RCA jacks.

USB Keyboard & Mouse

Finding a USB keyboard and mouse was pretty easy. But, if you don't have one, those keyboards are pretty cheap, but they are another small expense.

SD Memory Card

You will also need a SD card to store the operating system and your programs and data. It's a clever memory solution and SD cards are easy to get. The FAQ says "we have tried cards up to 32GB and most cards seem to work OK. You can also attach a USB stick or USB hard drive."

The default operating system for the Raspberry Pi is Debian, a version of Linux. The special "Raspbian wheezy" version of Debian, which is recommended for those just starting out. Once downloaded from the Raspberry Pi website, the wiki gave me instructions on how to copy the image to my SD card by hooking it up to a card reader and my Mac. There were 12 steps including typing commands in the terminal program like this: sudo diskutil unmount /dev/disk3s1 sudo dd bs=1m if=~/Downloads/debian6-19-04-2012/debian6-19-04-2012.img of=/dev/rdisk3

Using the device name of the partition work out the raw device name for the entire disk, by omitting the final "s1" and replacing "disk" with "rdisk" (this is very important: you will lose all data on the hard drive on your computer if you get the wrong device name).

A recent update to the wiki instruction page includes a "mostly graphical interface" method, compared to the original command line method, to copy the image to your SD Card. Still, you need to run the terminal and this command:

sudo dd if=path_of_your_image.img of=/dev/diskn bs=1m

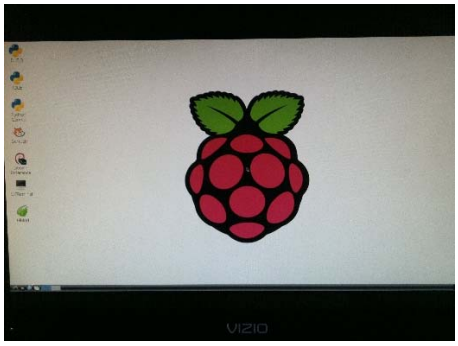
Once the operating system is loaded on the SD card, you can insert it into the slot on the Raspberry Pi and hopefully it will boot up.

Ethernet Connection

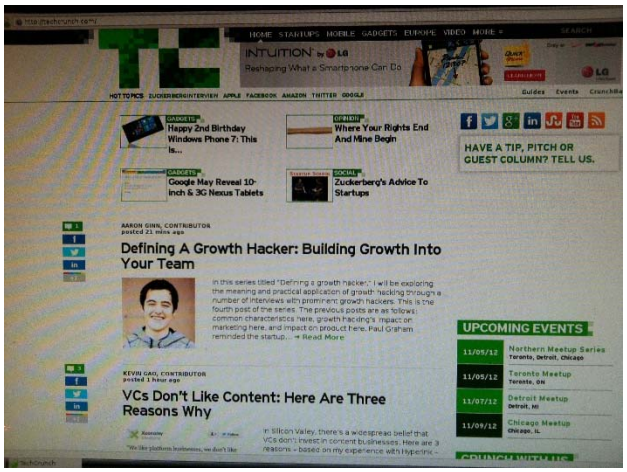
To get full use out of the Raspberry Pi, plugged it into an Ethernet LAN cable and had no problem getting online. You can also get it connected to the Internet using a USB Wi-Fi or wireless adapter.

Software & Browser

When the Raspberry Pi boots up, it will start in command line mode. launch its graphical user "X window system" by typing startx at the command line.



The operating system comes with the Midori web browser installed. Here's what the TechCrunch site looked like:



What's Next

The Debian operating system comes with Python pre-installed and it's the recommended default educational programming language. With just one command line of text, able to install PHP. You can run a web server or a media server from the Raspberry Pi. You can install WordPress, the software that powers TechCrunch, on it. You can blink a single LED or create an amazing 8x8x8 LED Cube. If you are a gamer, you can run Quake III,

As the Raspberry Pi team writes "We don't think that the Raspberry Pi is a fix to all of the world's computing issues; we do believe that we can be a catalyst. We want to see cheap, accessible, programmable computers everywhere."

After getting started with the Raspberry Pi, Raspberry Pi Foundation is a very worthy goal. And they are achieving this goal around the world today.

Humanoid Robots

CATHRINAL J
II year EEE

ARUNKUMAR C
II year EEE

Introduction

Time waits for none. Apparently, it's the same case with the speed of Technology upgrade these days. Moore's law is already broken. The happening innovation keeps us wondering 'how amazing it is' and 'what's next' at the same time. Since always, Advancements in technology are inspired by Science Fiction and action movies for implementing the fictional gadgets into reality.

What is a Humanoid Robot?

Humanoids are the robots which are designed as to look like humans. *Why Humans?* Well, robots are created by humans to reduce the repetitive manual work which was previously done by them. So why not design it like humans. Let us not forget, humans have evolved over time and again since millions of years in the first place. This *human type design* provides better grip for regular tasks, better awareness of obstacles and better interaction with humans. Most importantly, it is proved immensely helpful for biological advancements in the human body. You can find all kinds of robotic applications available from DIY to Integrated with our nervous system. The more it amazes you, the costlier it gets. If you go for most advanced, just a limb can cost you a hundred thousand dollars.

In the initial years, it was only thought for automating repetitive manual in the industrial sector. It was implemented and working fine for a long time now. After the advancements in biological engineering, hopes in the consumer sector gained momentum. Today, after the introduction of artificial intelligence (AI), the scenario has taken a drastic upside turn for the scope and advancements in this field. After the integration of AI, it is practically a Human being. It doesn’t only look like a Human being but it has its own thought process, own strategic thinking ability and although they may not be able to feel, but it can acknowledge the possible emotion as well.

Unmanned robots have already found their way into applications in our day to day lives. Recently, it made its debut as an Olympic Torchbearer in South Korea. Although most of these robots contribute to Floor Cleaning Robots, they have acquired their place in every segment of

tedious work done by humans. Now, as humans can be a major threat to mankind, Debates have come up with a swirl on whether AI can be a threat to mankind as well. These two have amalgamated so well that there's nothing wrong in saying that now the words 'Humanoid Robots' and 'Artificial Intelligence' go hand in hand.

Today, we have a wide range of Robots available in the market from ₹4000 to whatever you can pour in. However, the best one will be the one created by you yourself. It covers best of both worlds, Electronics and Computer engineering and gives you a product which is a peerless example of how innovation happens at the edge of two different domains. Let's see how basics are used to create one of the leading advancements in the world today.

Robots like Humans:

Today, we have Robotic to Human size ratio of 1:1. The latest arms are equipped with synchronous motors which are used for simultaneous movements with single instruction. But for machineries and repetitive manual work, most widely used structure is Six-axis Powerball arm. They are made of 3 Ball joints, 5 joints overall; one ball joint can be considered as a shoulder joint, another ball joint will replace the ankle while the last ball joint forms the wrist. The rest of the structure can be considered as upper and lower arm structure which provides strength to the arm structure. It is created in such a way that it can cover all the area in its outreach, unlike our hand. For example, you cannot bend your hand in the backward direction from the ankle but this structure can do turn in every direction; the only limit is its length.

Legs in a Humanoid Robot:

The joints are same as hand but instead of hand architecture, we can use hoverboard kind of architecture as it would be the most convenient one for its movement from one place to another.

Although it's lower body as well as upper body, especially the limbs are created as per the requirements and the purpose for what it's made and hence, Hoverboard architecture is rarely used for the lower limbs. They are been made to be same as humans internally as well as externally. The artificial Rib cage is used for its Torso part which gives protection to the internal circuit and mechanics. Legs are created to duplicate the maneuver just like human legs.

Other Devices in Humanoids:

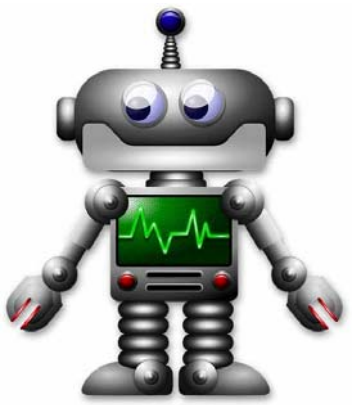
Other minimal device requirements for creating a Humanoid are HD Cameras, which will help collect data for Image processing, RFID, which will help give us instructions remotely, IR Sensors, to detect obstacle, humans, and distance from them, Rotors, for their movement, Accelerometer, Magnetometer, Gyroscope and Ethernet connection for controlling the movement if needed, A CPU, A Battery Unit, A PLC is must for regulated outputs and designing the logic.

Main Applications of Humanoids:

There are unlimited possibilities with robots and humanoids from doing a repeated manual task, to think and take a decision using Artificial Intelligence. Here I am mentioning few main areas where Humanoids are used:

Personal Home Assistants:

Integrating IoT with these robots will create a personal Home assistant which manages your Electricity, Light, Sound, and Temperature at your home using your home automation system.



Virtual Reality:

Integrating Virtual Reality with these robots and programming them to replicate our motion movements will give you a replica of yourself which will be useful in Fire, Flood, Experimentation or other dangerous situation where human life can be at stake. We can find all kinds of application of this futuristic bots but let's discuss the most important one. The one which will be the most helpful for the people in need: Prosthetics.

Prosthetics:

Prosthetics is one of the main reasons why it is made to look like actual body parts. Humanoids can be used a transplant or a replacement of a body part which will be connected to the nervous system of the body. A person without hand after elbow can make the hand customized for oneself to be connected to the nerves below the upper arm near the shoulder. Each nerve movement and extension of the inner part of the upper arm will give a unique signal to the brain. This signal will be corresponding to a specific hand movement; the modular prosthetic limb will recognize the muscle pattern and movement in the hand will take place accordingly. Not to forget that each of these things will happen in real time. Just by the trigger of a thought, you can move the hand just like it is your own hand. It will change the way special people live. It will be helpful for the people who went

through an accident and lost a limb or other body part. The best technology available today also has over 100 sensors in the palm itself, providing sensation capabilities of an actual touch on the surface of the artificial arms and legs which provides successful targeted sensory innervation and feel.

Mind Controlled Prosthetic Arm developed by DEKA Research & Development Corp

The creation and advancements of this Humanoid Robots are never-ending. Major innovations and breakthroughs happen every day in this very field. Numerous applications from industrial to now in our day to day life direct the trend of these droids to keep up and turn upwards. Multiple jobs will be replaced by robots in near future. This issue is debated in the parliaments around the world. The idea of *'Basic income for everyone'* was introduced as one of the options while keeping the recessions and unemployment in mind. The government of several countries has shown resistance and placed their thoughts about putting a ban on AI while some of the governments showed interest in restriction of the applications of humanoid robots and them replacing the jobs.

The importance of these products can be easily seen with the surge of the issues, merits, and demerits being discussed in the houses. One thing is for sure, with the mixture of best in the class software of Artificial Intelligence and these best in class physical structures of circuitry, the product which is known is going to take over the world very fast even with any restrictions. And the development of these kinds of Humanoids has already started, as we can see the Sophia Robot as the great example of Artificial Intelligence with perfect human physical structure.

ZigBee

MANIKANDAN S
IV year EEE

GOVARTHANAN E
IV year EEE

Introduction

Generally many people get confused with two terms XBee and ZigBee, most of them use it interchangeably. But this is actually not the case; ZigBee is standard protocol for wireless networking. While XBee is a product that supports various wireless communication protocol, including ZigBee, Wi-Fi (Wi-Fly module), 802.15.4, 868 MHz module etc. Here we are mainly focused on Xbee/Xbee-PRO ZB RF module which consists of ZigBee firmware.

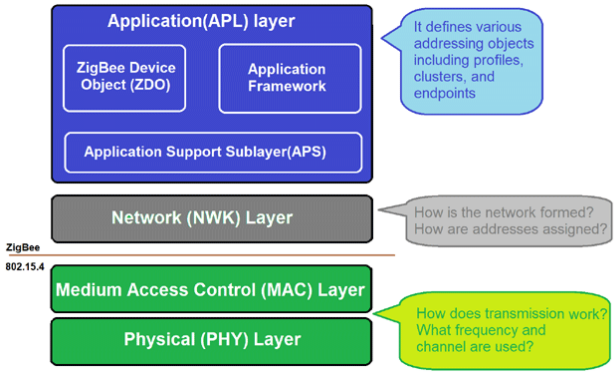
Just think of a calculator in computer, where complex calculations are performed with user friendly interface. The task would have been very difficult and tedious if only hardware would have been available. So, at highest level, the availability of software makes problem solving process easier. Whole process is divided into layers of the software by the actual hardware which is called by higher levels.

For understanding first term just go by its name, “route” which means to track or identify the path. In networking, routing means to provide direction to the data from source node to destination node. When two nodes in network attempt to transmit simultaneously, creates a situation called collision. So, generally *Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) technique* to avoid collision you can learn more about CSMA using this link. Basically in it the nodes talk in the same way that human conversation; they briefly check to see that no one is talking before they start to send data.

Whenever receiver successfully receives the transmitted data, it acknowledges the transmitter. The flow of data must not be allowed to overwhelm the receiver radio. Any receiving radio has a limited speed at which it can process incoming data and a limited amount of memory in which to store incoming data.

ZigBee Architecture:

There are major four layers available in ZigBee stack which are physical layer, Media access layer, Network layer and application layer.



Application layer defines various addressing objects including profiles, clusters, and endpoints. You can see the ZigBee stack layers in the figure above.

Network layer: It adds routing capabilities that allows RF data packets to traverse multiple devices (multiple "hops") to route data from source to destination (peer to peer).

MAC layer manages RF data transactions between neighboring devices (point to point). The MAC includes services such as transmission retry

and acknowledgment management and collision avoidance techniques.

Physical layer: It defines how devices are connected to make a network; it defines the output power, number of channels and transmission rate. Most ZigBee applications operate on the 2.4 GHz ISM band at a 250kbps data rate.

Most XBee families have flow control, I/O, A/D and indicator lines built in which can be configured using appropriate commands. Analog samples are returned as 10-bit values. The analog reading is scaled such that 0x0000 represents 0V, and 0x3FF = 1.2V. (The analog inputs on the module cannot read more than 1.2V) To convert the A/D reading to mV, do the following:

```
AD (mV) = (A/D reading * 1200mV) / 1023
```

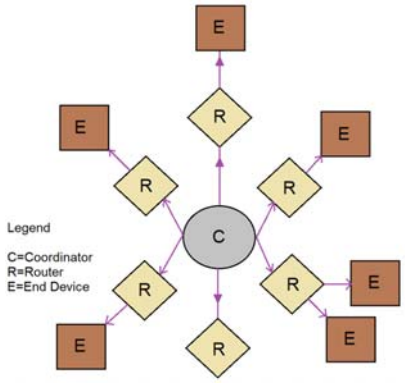
Data Transmission in ZigBee

You can call a network as combination of software and hardware which is capable of sending data from one location to another. Hardware is responsible for carrying the signals from one point of network to another. Software consists of instruction sets that make it possible to work as we expect.

Generally the data transmission by ZigBee packets can be done in two ways: unicast and broadcast.

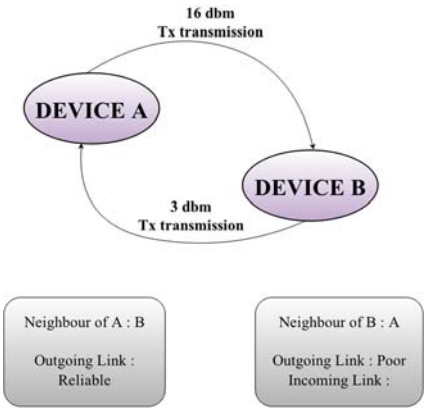
Broadcast Transmission:

In simple words Broadcast means the information/programme transmitted by radio or TV. In other words broadcast transmissions are sent to many or all devices in the network. Broadcast transmissions with the **ZigBee protocol** are propagated in the whole network such that all nodes receive the transmission. To accomplish this, the coordinator and all routers that receive a broadcast transmission will retransmit the packet three times.



Unicast Transmission:

Unicast transmissions in ZigBee route data from one source device to another destination device. The destination device could be an immediate neighbor of the source device, or it could have several hops in between the way. An example is shown below in the figure explaining mechanism for recognizing the reliability of the bi-directional link.



Basics of network for Xbee routers and Coordinator

To reach your friend's house, what you need? You just need his address. Similarly, for sending the data from one Xbee module to other, you need its unique address. Just like with people, Xbee even have several addresses, each one have a particular role in networking. There are two types of

addresses Static address (64-bit address) and Dynamic address (16-bit address).

Addresses:

64-bit address is unique universally; it is firmed inside the Xbee module by the manufacturer. No other ZigBee radio on earth will have that same static address, on back of every xbee module you can see this address as shown below, and notably the higher part of address "0013A200" is same for every xbee module.

A device receives a **16-bit address** which should be unique locally, when it joins a ZigBee network. The 16-bit address 0x0000 is reserved for the coordinator. All other devices receive a randomly generated address from the router or coordinator device that allows the join. The 16-bit address can change when two devices are found to have the same 16-bit address or a device leaves the network and later joins (it can receive a different address).

Node Identifier:

It is always easier for our brain to remember strings instead of number. Hence, each Xbee module in a network can be assigned with a node identifier. **Node identifier** is set of characters i.e. strings which can be more human friendly way of addressing a node in a network.

Personal Area Networks:

Network developed by these Xbee modules are called **personal area networks or PANs**. Each network is defined with a unique PAN identifier (PAN ID). This identifier is common among all devices of the same network. ZigBee supports both a 64-bit and a 16-bit PAN ID. Both PAN addresses are used to identify a network uniquely. Devices on the same ZigBee network must share the same 64-bit and 16-bit PAN IDs. If multiple ZigBee networks are operating within range of each other, each should have unique PAN IDs.

The 16-bit PAN ID is used to address MAC layer in all RF data transmissions between devices in a network. But, due to the limited addressing space of the 16-bit PAN ID (65,535 possibilities), there may be chance that multiple ZigBee networks (within range of each other) can have the same 16-bit PAN ID. To resolve these conflicts, the ZigBee Alliance created a 64-bit PAN ID. ZigBee defines three different device types: coordinator, router, and end device.

One coordinator is always required in every network for charging of setting up the network. So, it can never sleep. It is also responsible for selecting a channel and PAN ID (both 64-bit and 16-bit) to start the network. It can allow routers and end devices to join the network. It can assist in routing data in a network.

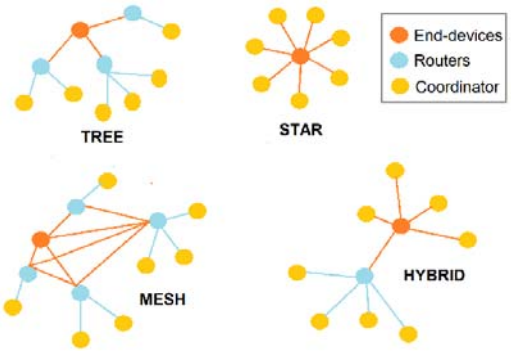
There can be multiple routers in a network. One router can get signals from other routers/EPs (End Points). It can also never sleep. It must join a Zigbee PAN before it can transmit, receive, or route data. After joining, it can allow routers and end devices to join the network. After joining, it can also assist in routing data. It can buffer RF data packets for sleeping end devices.

There can be multiple End Points as well. It can go in sleep mode to save power. It must join a ZigBee PAN before it can transmit or receive data and it

cannot even allow devices to join the network. It is dependent on parent for transmit/receive data.

Since the end device can go in sleep mode, the parent device must buffer or hold incoming data packets until the end device wakes up and receive the data packets.

Different Network Topology in ZigBee



In **Mesh Topology**, every node is connected with each other node expect the end device because end devices can't communicate directly. To enable simple communication between two ZB radios, you'll need to configure one with the coordinator firmware, and one with router or endpoint firmware. Main advantage of Mesh network is that if one of the links becomes unusable, it does not incapacitate the entire system.

In a **star topology**, each device has a dedicated point-to-point connection to a central controller (Coordinator). All the devices are not directly linked to each other. Unlike a mesh topology, in star topology one device can't send anything directly to another device. The coordinator or hub is there for exchange: If one device wants to send data to another, it sends the data to the coordinator, which further sends the data to the destination device.

Hybrid network are those networks which contains two or more types of communication standards. Here, hybrid network is combination of star and tree network, few end devices are connected directly to the coordinator node and other end devices needs the help of parent node to receive the data.

In **Tree network**, routers forms the backbone and end devices generally clustered around each router. It's not very different from a mesh configuration except the fact that there routers are not interconnected you can visualize these networks using figure shown above.

XBee AT Commands:

AT (TEST): This is the test command to check if the module is responding an OK as reply confirms the same.

ATDH: Destination Address High. To configure the upper 32 bits of the 64-bit destination address DL and DH combined gives you 64 bit destination address.

ATDL: Destination Address Low. This again for configuring the lower 32 bits of the 64-bit destination address.

ATID: This command changes the PAN ID (PersThe ID is 4 bytes of hexadecimal and can range from 0000 to FFFF

ATWR: Write. Write parameter values to non-volatile memory so that parameter modifications persist through subsequent resets.

Note: Once WR is issued, no additional characters should be sent to the module until

After the "OK\r" response is received.

ATRE (Restore Defaults): Restores factory settings to the module, is very useful if the module does not responds.

3D Printing

SATHIK NIVAS J
IV year EEE

AJITHKUMAR K
IV year EEE

Introduction

3D printing or additive manufacturing is a process of making three dimensional solid objects from a digital file. It uses a printer like machine which melts a material (commonly plastic) and pours it on a base in a predefined manner thus creating successive layers to make it 3 dimensional. To put it simple the printer builds a loaf of bread slice by slice.

There are many types of 3D printing technologies available but the most used is Stereolithography (SLA) because it is more affordable compared to others.

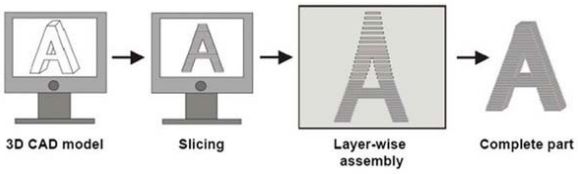
This machine is capable of converting any 3D designed file into a 3D object by the following steps.

1. A 3D file is checked by software to see if it is printable.
2. The software then slices the file and sends it to a machine.
3. A plastic filament is fed through the printing head.
4. The filament melts and the head then deposits it on a build platform.
5. The head moves around depositing the material so that it makes up the bottom slice of the object.
6. The head returns to its starting position and the build platform is lowered.
7. The head then builds the new layer on top of the first and this is repeated until the object is done.

Get started with 3D printers

Since the recent technology has reduced the price of 3D printers low enough for an average person to buy and use it, 3D printing is no more a magic or Industrial Level equipment. Thanks to the online forums and Designs now anyone who can operate a computer can print his imagination. Just like how laser printers have become common in our house these 3D printers are yet to find their way into our house or workplace. So let us see how one can **get started with 3D printing**, for any print to take place the following are the four simple steps to be followed:

1. 3D CAD modelling
2. Slicing and other settings.
3. Layer - wise printing
4. Complete part



3D CAD Modelling:

Yes, in order to print something with your printer you have to design them modelling software.

If you are a **professional designer** you can use any of your modelling software that has the ability to save a file in .STL(Stereolithography) or .OBJ(object). Once the design is created is can be taken to the slicing stage. Here you are free to alter, edit or tweak your desing the way you like since the complete design is made by you.

If you are a **Learner,** then there are tons of free open source software available which can be used the design in your mind. It would just be like placing building block. If your design is too good you will also paid by some online communities.

If you are a **newbie**, and has no idea how to design a 3D model Worry not!.. There are lots of websites which provide free STL files. These files are tested and printed by persons who designed it hence it is reliable and will be easy to print. The only limitation to these files is that they come in .STL formats (Mostly). This format cannot be customized the way we like but even then it can be resized with the help of slicing software. If you are an absolute beginner then we would recommend you by starting with some designs available in the market.

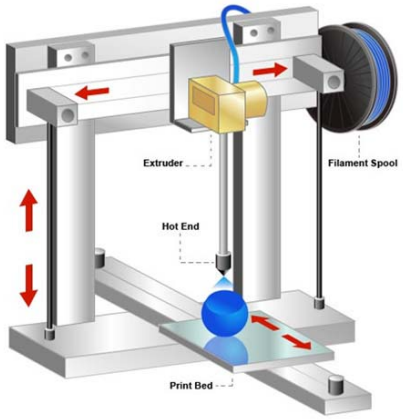
Once the design is made the work of 3D modelling software is over. The files are now sent to Slicing software where it gets ready for the printer to print it.

Slicing and Other Settings:

The models that are designed have to be converted into a series of code called the G-Code for the printer to recognize it and start printing. This part is called Slicing. The quality of the print depends on the settings that are made here. Before we jump into this lets us get familiar of how simple a 3D printer really is!!!!

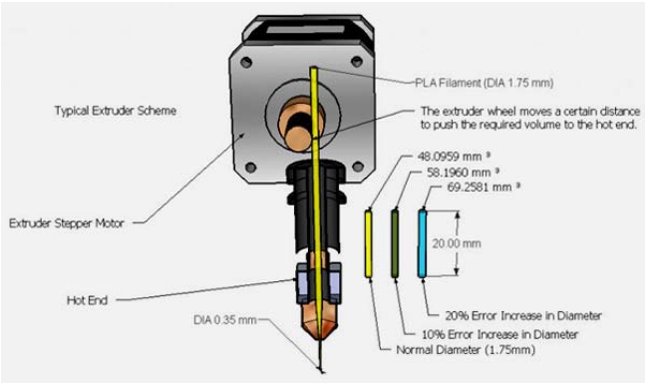
Working of 3D printer

The typical parts of a 3D printer are shown in the picture below.



The **Extruder** is the most important part in our printer. This is the place where the filaments gets melted and is extruded. The extruder consists of narrow tube through which the filament is pushed by a Stepper motor. It consist a heating material commonly called as **hot end** which heats the filament, the heated filament is pushed outside the nozzle similar to pressing toothpaste out of its nozzle. As the filament melts oust the stepper motor will push the filament inside which will be mounted as a spool on a **spool holder.**

There many types of filament and a lot more come into the market every month. Each filament has its diameter and printing temperature, the most used one PLA which has temperature of 180-210 deg C.



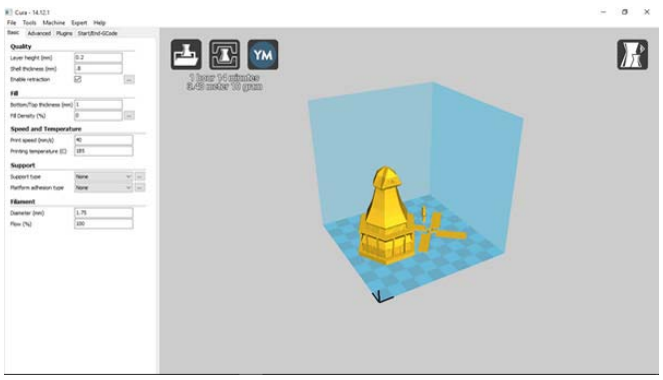
This complete extruder set up is mounted over a movable rails. As the filament melts and comes out the extruder gets moved by other stepper motors (in the direction of the red arrows) present in the required direction and prints on the **Base Plate**.

Now, that we have some basic Idea of how our Printer works, let us get back to our **SLICING software.** The most used software is Cura this software was mainly created for the printer called Ultimaker. But since it was made free and open source most of the printers out there have started using Cura.

In this software the print settings are sent just like we do for our laser printer. Here we can Scale, align, and orient our print model. Other crucial settings like the temperature, retraction speed, flow of the filament nozzle diameter etc are fed to the printer using this Software.

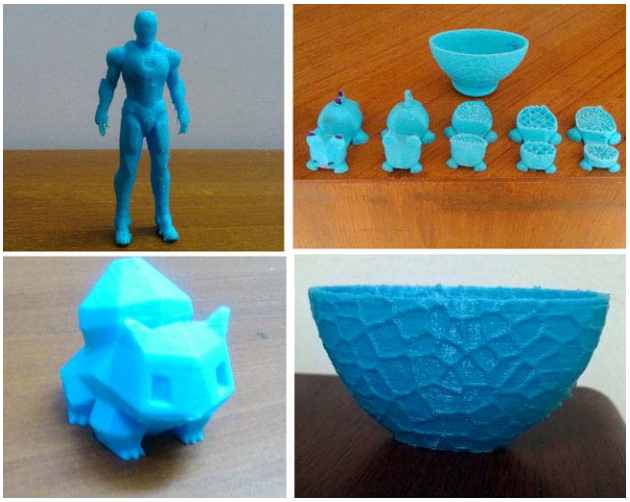
Layer - wise printing:

Based on the settings the Print time and material length required for the print will be displayed by the software itself, as shown in the picture below. Here the print time is 1 hour 14 minutes and the material used will 10 gram. The other settings for this print can also be found in the picture



Once the STL file is loaded and settings are done, this software can be used to visualize how our print will be made using the layers option, as shown in the picture below. This how or print will look like when it printing the 78th layer of this print. The blue color line indicates the path of our extruder.

Once this is done the file can be converted into a G-Code which can be fed to the printer for printing. That is it, you model will be printed in the time mentioned.



Conclusion

3D printing however does have its own set of challenges. It suffers from slow operating speed, mechanical defects in products and limited material choices (mostly Plastic). Due to slow operating speed of 3D printers, manufacturing companies might lose out to competitions with inefficient go to market strategy. But hopefully this field is evolving as you are reading this article, and has a great scope ahead. It all depends on how one decides to use this technology.

Motion Sensors

ANITHA M
IV year EEE

HARINI K
IV year EEE

Introduction:

We are surrounded by all kinds of sensors in this modern and magical age we live in. Consider the many sensors in a late-model automobile: The engine compartment has crankshaft and camshaft angle sensors, the seats have pressure sensors (to determine if a child or an adult is occupying the seat) and seat belt sensors, and of course life-saving crash sensors are used to inflate airbags in the event of a collision. It is estimated that current model cars have between 60 and 100 sensors per car, and newer (smarter) cars will have close to 200 sensors.

Ever wonder how your smartphone knows when to rotate its screen—from landscape to portrait—depending on its orientation? You can thank accelerometers for that magic trick. These same accelerometers get credit for allowing you to play games on your phone that require movement of the phone itself. While accelerometers are the actual gravity sensors, both magnetometers and gyroscopes play a role by removing linear acceleration from the accelerometer data.

PIR

The PIR, or passive infrared, detector is commonly used in intruder alarm systems. A PIR is similar to a thermal sensor. These infrared sensors/detectors are the secret sauce inside motion sensor security lights that illuminate driveways when someone approaches. Infrared is outside our eyes' light-detecting abilities

PIRs are pyroelectric devices—i.e., they can generate a voltage when heated or cooled—that detect motion by sensing changes in the radiant heat (aka infrared) emitted by surrounding objects.

Figure 2. *PIR sensor.*

PIR sensors typically include a field-of-view and/or a linear distance specification, and most come with a plastic lens. Figure 2 below is an example of a PIR sensor.

Accelerometer

As mentioned previously, accelerometers are inside smartphones (and tablets) resulting in accelerometers being low-cost and readily available given the ultra-mass production of smartphones.

Many accelerometer devices use MEMS (microelectromechanical systems) as the critical technology for sensing acceleration. STMicroelectronics accelerometers use a proprietary process that allows suspended silicon

structures to freely move in the direction of the sensed acceleration. According to STMicroelectronics, "When an acceleration is applied to the sensor the proof mass displaces from its nominal position, causing an imbalance in the capacitive half-bridge. This imbalance is measured using charge integration in response to a voltage pulse applied to the capacitor"

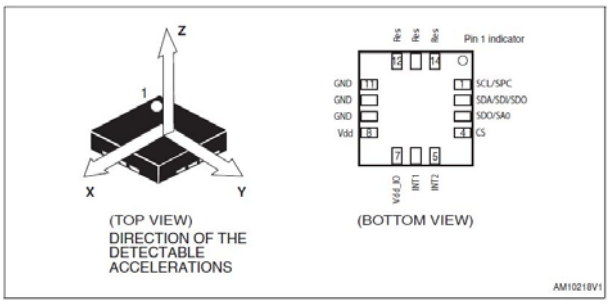


Figure 3. *3-axis accelerometer.*

Tilt Sensors

Somewhat similar to the accelerometer, but far less complicated in design, is the tilt sensor. The tilt sensor, also referred to as a motion detector or a tilt switch, is exactly as it sounds: it's a sensor that detects tilting, specifically the tilting from the horizon. There are multiple tilt sensing design types, but a common one is called the rolling ball sensor switch (see Figure 4). The rolling ball sensor switch works by means of a rolling ball which, when tilted to a critical point, either makes or breaks contact with two or more conducting pins.

Tilt sensors' specifications normally include the sensor angle, which is given in units of degrees.

Although not commonly used today, the original type of tilt switch—the mercury tilt switch—can still be found, if you look hard enough. If you find a tilt switch that is RoHS compliant, rest assured that it will not contain mercury.

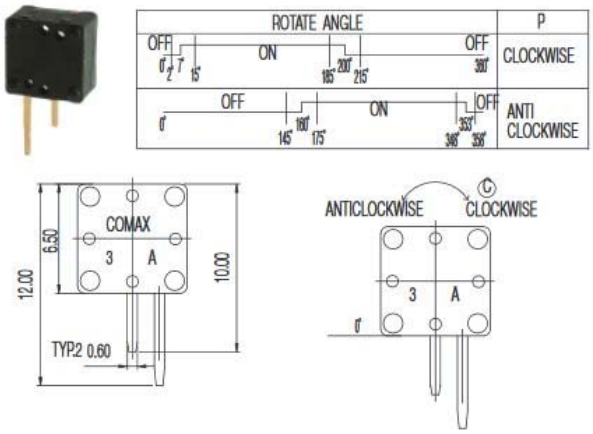


Figure 4. *Rolling ball sensor switch*

Vibration Sensors

There are many different methods and sensors used for measuring and detecting vibrations. One such sensor is the aforementioned accelerometer. Many digital accelerometers have user-selectable sensitivity settings which allow the devices to detect accelerations ranging from very large to very small.

The piezoelectric sensor (see Figure 5) is ideal for vibration sensing technologies. A piezoelectric sensor uses the piezoelectric effect to measure changes in acceleration, strain, or force by converting the physical change to an electrical change. In addition to detecting vibrations, these same types of sensors are used for detecting mechanical shock.

Figure 5. *A piezoelectric vibration sensor.*

Rotational Sensors

Perhaps the most simple rotational sensor is using a potentiometer as a voltage divider where the voltage is proportional to the angle of rotation.

Quadrature encoders, commonly used for measuring rotational position, provide a specific number of equally spaced pulses per revolution. A quadrature encoder is a type of incremental encoder. Incremental encoders provide relative position feedback (and can also provide speed and direction) by generating a stream of binary pulses that are proportional to the rotation of a motor's shaft. Quadrature encoders have two channels that are 90 electrical degrees out of phase with one another. See Figure 6.

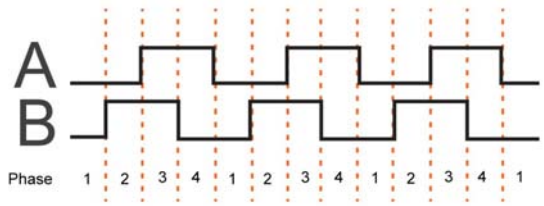


Figure 6. *Quadrature encoder's binary pulses.*

Other rotational sensors include hall-effect-based rotational sensor ICs and rotary position sensors.

Force Sensors

There are many types of force sensors available, two of which are strain gauges and load cells.

A load cell converts the deformation of a material, measured using a strain gauge, into an electrical signal. One type of a load cell uses a bending beam. Let's assume a strain gauge is mounted on the beam. As a force is applied to the beam the beam begins to slightly bend resulting in bending the strain gauge, leading to a change in its electrical output signal. See Figure 7 for an example of a load cell.

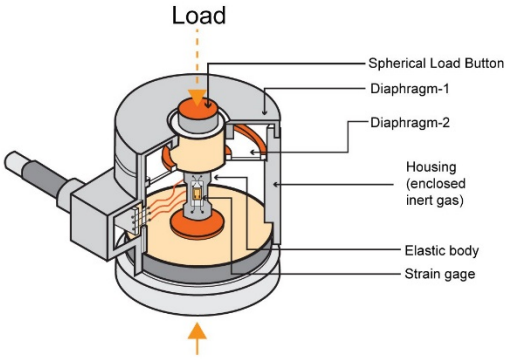


Figure 7. *Load cell depiction.*

Pressure Sensors

There are many different types of pressure sensors and technologies used to make pressure sensors.

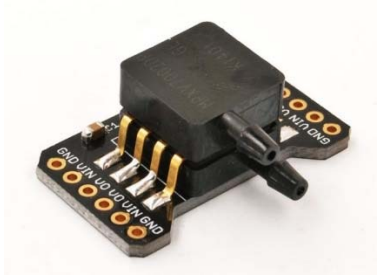


Figure 8. *Differential pressure sensor.*

Types of pressure sensors:

- Absolute pressure sensor: This sensor measures a pressure relative to a perfect vacuum.
- Gauge pressure sensor: This type of pressure sensor measures a pressure relative to the atmospheric pressure.
- Sealed pressure sensor: This sensor measures a pressure relative to some other known and fixed pressure.
- Differential pressure sensor: This pressure sensor measures the difference between two pressures. The sensor "sees" each of the two pressures at the same time.

C for Embedded Applications

NEERAJ P
IV year EEE

PONKAVIN T
IV year EEE

Introduction

By the standards of modern technology, C is a rather old language. The original development took place in the early 70s, followed by revisions in the late 70s and standardization in the 80s. Nevertheless, in my opinion it has lost none of its vigor. It's still a great language for embedded applications, and in my experience it is a suitable programming environment for everything from simple microcontroller-based devices to sophisticated digital signal processing.

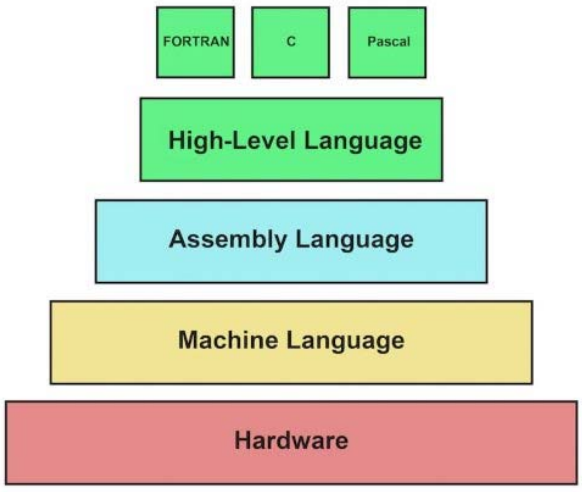
The Need for C

There are at least a few electrical engineers who do not know how to write a program in C and never will need to write a program in C. If you're the sort of person who prefers hardware to software, you might consider these individuals "the lucky ones."

Whether we like it or not, though, programming is an increasingly important part of electrical engineering, and actually I have found much satisfaction in being able to not only design circuit boards but also write the firmware for those boards. These two aspects of system development are closely related, and I suspect that the end result is often superior when board design and firmware development are carried out by the same person.

C vs. Assembly

Writing firmware in assembly is slow and error-prone, and maintaining an adequate level of organization in long, complex programs is hopelessly difficult. You cannot really understand high-level languages if you don't understand assembly. If you've never had the opportunity to gain some solid experience with assembly language, you should at least familiarize yourself with some of the basic concepts before you dive into C. The articles listed above in the Supporting Information section are a good place to start.

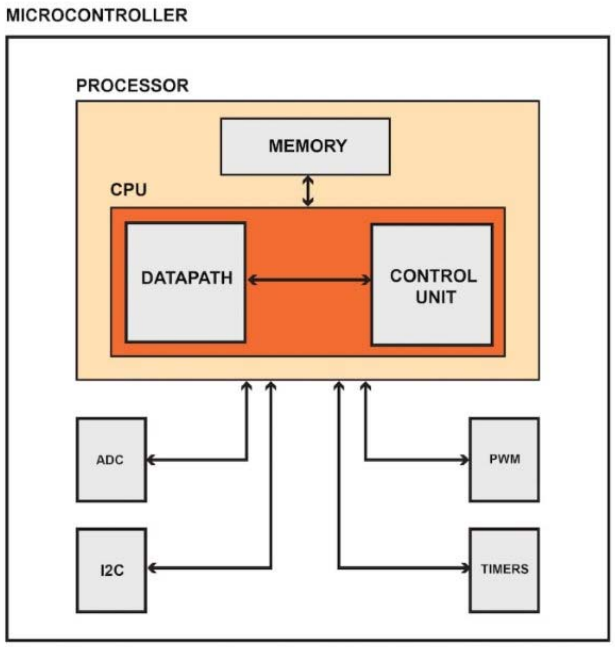


What Do Processors Understand?

Only machine language. Ones and zeros. All of the "programmer-friendly" aspects of the C language must eventually be translated into the low-level reality of the processor's digital hardware—i.e., binary

arithmetic, logical operations, data transfer, registers, and memory locations.

It certainly is possible to successfully write a program in C while knowing nothing about the actual hardware, but in the context of embedded systems development, it's helpful and sometimes necessary to understand both your hardware and how your C code interacts with that hardware.



The Basics

C programs range from those that are quite simple to those that are very complex. In the embedded world, many programs will tend toward the simple side of the spectrum, and the basic programming elements described below provide a good foundation for further study of C-language firmware development.

Include Statements

An embedded C program will begin with at least one #include statement. These statements are used to introduce the contents of a separate file into your source file. This is a handy way to keep your code organized, and it also allows you to use library functionality, hardware-configuration routines, and register definitions provided by the manufacturer.

The code excerpt below shows the include statements that I used in one of my microcontroller projects. Note that the "Project_DefsVarsFuncs.h" file is a custom header file created by the programmer (i.e., me). I was using it as a convenient way to incorporate preprocessor definitions, variables, and function prototypes into multiple source files.

```
//-------------------------------------
-----------------------------------------
// Includes
//-------------------------------------------------
-------------------------
#include                // SFR declarations
#include "Project_DefsVarsFuncs.h"
#include "InitDevice.h"
#include "cslib_config.h"
#include "cslib.h"
```

Pre-processor Definitions

You can use a #define statement to create a string that will be replaced by a number. Preprocessor definitions are not necessary, but in some situations they are extremely helpful because they allow you to easily modify a value that appears in various different portions of your program.

For example, let's say that you're using the microcontroller's ADC and that your code uses

the ADC's sample rate in several separate calculations. A preprocessor definition allows you to use an intuitive string (such as SAMPLE_RATE) instead of the number itself in the calculation code, and if you're experimenting with different sample rates, you only need to change the one numerical value in the preprocessor definition.

```
#define SAMPLE_RATE 100000
```

You can change 100000 to any other number, and this new number will be used to replace all instances of the string SAMPLE_RATE.

Preprocessor definitions are also a great way to make code more readable. The following is a list of handy #define statements that I incorporate into all of my firmware projects.

```
#define BIT7 0x80
#define BIT6 0x40
#define BIT5 0x20
#define BIT4 0x10
#define BIT3 0x08
#define BIT2 0x04
#define BIT1 0x02
#define BIT0 0x01
#define HIGH 1
#define LOW 0
#define TRUE 1
#define FALSE 0
#define SET 1
#define CLEARED 0
#define LOWBYTE(v)   ((unsigned char) (v))
#define HIGHBYTE(v)  ((unsigned char) (((unsigned int) (v)) >> 8))
```

It's important to understand that preprocessor definitions have no direct relationship to hardware. You're just telling the preprocessor to replace one string of characters with another string of characters before the program is compiled.

Variables

Processors store data in registers and memory locations. There really is no such thing as a variable as far as the hardware is concerned. For the programmer, though, writing code is much easier when we can use intuitively named variables instead of memory addresses or register numbers.

Compilers can manage the low-level details associated with variables without much input from the programmer, but if you want to optimize your use of variables you'll need to know something about the device's memory configuration and the way in which it handles data of different bit widths.

The following code excerpt gives an example of variable definition. This was written for the Keil Cx51 compiler, which reserves one byte of memory for an "unsigned char" definition, two bytes for an "unsigned int" definition, and four bytes for an "unsigned long" definition.

```
unsigned long
Accumulated_Capacitance_Sensor1;
unsigned long
Accumulated_Capacitance_Sensor2;unsign
ed int Sensor1_Unpressed;
unsigned int Sensor2_Unpressed;
unsigned int Sensor1_Measurement;
unsigned int Sensor2_Measurement;
unsigned int AngularPosition;
```

```
unsigned int TouchDuration;
unsigned char CurrentDigit;
unsigned int CharacterEntry;
unsigned char DisplayDivider;
```

Operators, Conditional Statements, and Loops

The core of computational functionality consists of moving data, performing mathematical computations and logical operations with data, and making programmatic decisions based on the value of stored or generated data.

Mathematical operations and bit manipulation are accomplished by means of operators. C has quite a few operators: equals (=), addition (+), subtraction (-), multiplication (*), division (/), bitwise AND (&), bitwise OR (|), and so forth. The "inputs" to an operator statement are variables or constants, and the result is stored in a variable.

Conditional statements allow you to perform or not perform an action based on whether a given condition is true or false. These statements use the words "if" and "else"; for example:

```
if(Sensor1 < Sensor2 && Sensor1 < Sensor3)
        return SENSOR_1;
else if(Sensor2 < Sensor1 && Sensor2 < Sensor3)
        return SENSOR_2;
else if(Sensor3 < Sensor2 && Sensor3 < Sensor1)
        return SENSOR_3;
else
return 0;
```

For loops and while loops provide a convenient means of repeatedly executing a block of code. These types of tasks arise very frequently in embedded applications. For loops are more oriented toward situations in which a block of code must be executed a specific number of times, and while loops are handy when the processor should continue repeating the same block of code until a condition changes from true to false. Here are examples of both types.

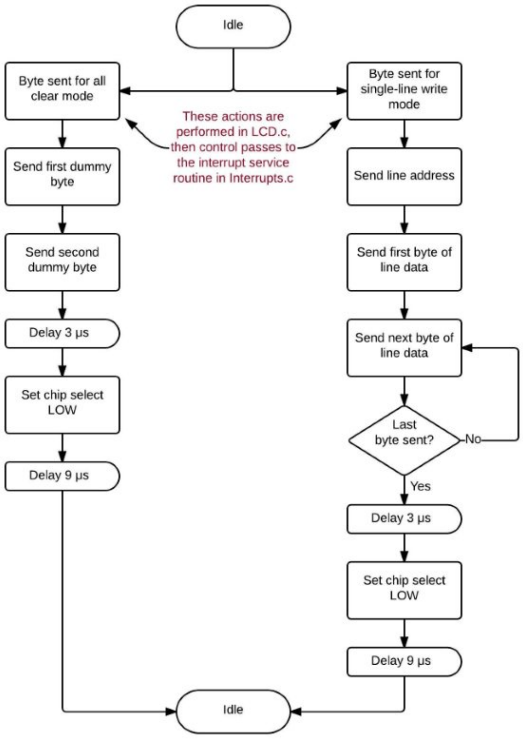
```
        for (n = 0; n < 16; n++)
{
        Accumulated_Capacitance_Sensor1 += Measure_Capacitance(SENSOR_1);
        Delay_us(50);
        Accumulated_Capacitance_Sensor2 += Measure_Capacitance(SENSOR_2);
        Delay_us(50);
}

 while(CONVERSION_DONE == FALSE);
{ LED_STATE = !LED_STATE;
Delay_ms(100);
}
```

Functions

Good C code is vastly superior to assembly code in terms of organization and readability, and this is due in large part to the use of functions.

Functions are blocks of code that can be easily incorporated into other portions of code. Causing the processor to execute the instructions contained in the function is referred to as "calling" the function. A function can accept one or multiple inputs, and it can provide one output, called a return value.



C's functions, conditional statements, and loops make it fairly easy to translate a flowchart into working code. This example is from a project in which I used SPI communication to control an LCD.

The use of functions does involve some overhead, so we have to be careful to not burden the processor with an excessive number of function calls, but in general the benefits of functions far outweigh the costs.

Here is an example of a function that has three numerical inputs and uses these inputs to generate a true-or-false return value.

```
bit Is_In_Range(int input, int LowerBound, int
UpperBound)
{
        if(input >= LowerBound && input
<= UpperBound)
                return TRUE;

        else
                return FALSE;
}
```

Conclusion

A thorough discussion of the C language could go on almost indefinitely, and this article has only scratched the surface. I hesitate even to publish something that omits so much important information, but we have to start somewhere. We plan to publish quite a few more articles on the use of the C language in embedded applications, and we'll fill in this introductory article with links as other resources become available.

Switches

ELANCHEZHIYAN P
II year EEE

DICKMAN PIOUS P
II year EEE

Introduction

Switch is nothing but a device which is used to turn ON and OFF the equipment. Most probably this equipment is electrical equipment like fan, TV etc. To flow current from a circuit it must require a close path (loop). If switch is OFF, that means circuit is open and current cannot flow through the conductor and equipment is de-energise (OFF state). To make it energise, we have to turn ON switch, it makes a complete circuit and close path. So, current can flow through the equipment and it can turn ON. So, function of switch is to make (switch is ON) and break (switch is OFF) the circuit.

In control system engineering, switches play an important role. **There are mainly two types of switches- mechanical switch and electrical switch**. Mechanical switches require physical or manual contact with switch for operation. Electrical switches not require physical or manual contact, it has ability to perform operation. Electrical switches operate under the action of semiconductors.

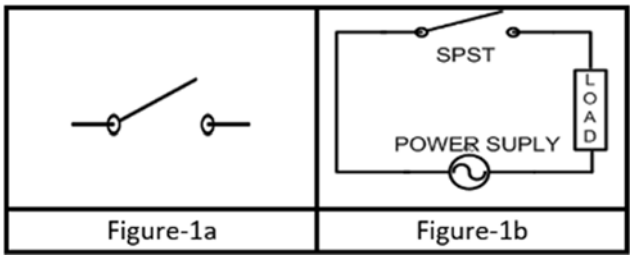
Mechanical Switches:

Mechanical switches further classify into **different types of switches** based on number of poles and throughs. Poles means the number of input circuit (power circuit) available to the switch. Throws means the number of output circuit (number of path in which current can flow) available to the switch.

1. Single pole single throw (SPST)
2. Single pole double throw (SPDT)
3. Double pole single throw (DPST)
4. Double pole double throw (DPDT)
5. Two poles six throw (2P6T)
6. Momentary operation switch / Momentary control switch
    1. Push button
    2. Pressure switch
    3. Temperature switch
    4. Toggle switch
    5. Rotary switch

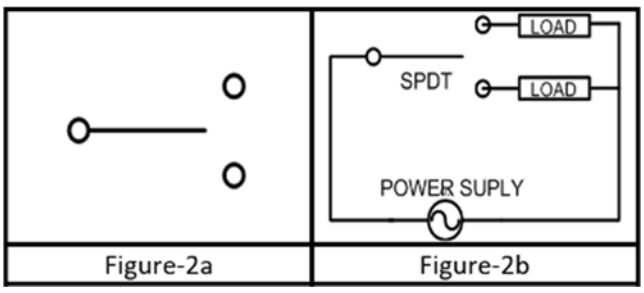
In **mechanical switch, two metal plates are touch each other to complete circuit** for the current to flow and separate each other to open circuit for the current to interrupt.

1) Single pole single throw (SPST): This switch consists of two terminals; one input terminal is known as pole and one output terminal is known as throw. So, name of this switch is ***single pole single throw.*** This switch is simplest example of switch. Generally, this switch used in single loop, means circuit requires to control only one close path. Symbol of single pole single throw switch is as shown in figure-1a. This switch is connected in series with the equipment, source or elements as shown in figure-1b.

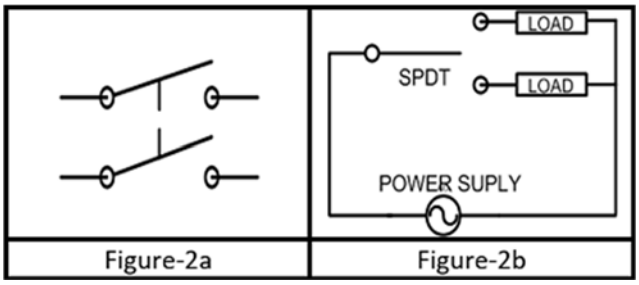


Working and symbol of SPST

2) Single pole double throw (SPDT): This switch consists of three terminals; one input terminal (pole) and two output terminal (throw) as shown in figure-2a. By using this switch, we can supply current or signal to two loops as shown in figure-2. Sometimes this switch is known as **selector switch**.

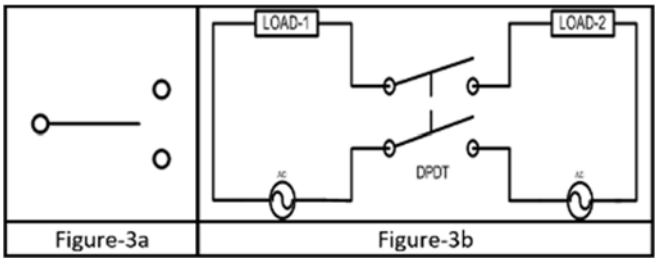


Working and symbol of SPDT Type 1



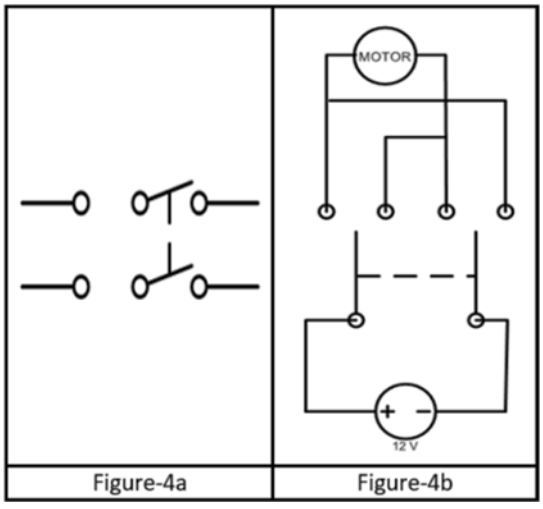
Working and symbol of SPDT Type 2

3) Double pole single throw (DPST): This switch consists of four terminals; two input terminal (pole) and two output terminal (throw) as shown in figure-3a. This switch is very similar to two SPST switches. Both switches are connected with single liver so, both switches operate at a single time. These switches used when we want to control two circuit for same time as shown in figure-3b.



Working and symbol of DPST

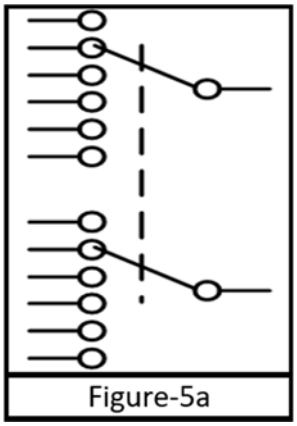
4) Double pole double throw (DPDT): This switch consists of six terminals; two input terminals (pole) and two terminals for each pole, so total four output terminal (throw) as shown in figure-4a.



Working and symbol of DPDT

Operation of this switch is similar to the two separate SPDT switches operate at a same time. In this switch, two terminal of input (pole) are connected with one set (two) of output (throw-1) in position-1 of switch. If we change the position of switch, it will connect this input with second set of output (terminal-2) as shown in figure-4b. Here as shown in example, let us assume that, in position-1 if motor is rotating in clockwise direction, if we change to position-2 motor will rotate in anti-clockwise direction.

5) Two poles six throw (2P6T): This consists of fourteen terminals; two input terminals (poles) and six terminals for each pole, so total twelve output terminal (throw) as shown in figure-5a. Generally, this **type of switch** is used for change-over in circuit with common input terminal.



Symbol of 2P6T

6) Momentary operation switch:

1. ***Push button switch:*** when you press the switch, contacts of switch is closed and make circuit close to flow the current and when you remove pressure from the button, contacts of switch is open and break the circuit. So, this switch is momentary contact switch which is able to control the circuit by making and breaking its contact. In push button switch, when you remove pressure from the switch, there is an arrangement of spring to open contact.
2. ***Pressure switch:*** This type of switch consists of C-shape diaphragm. According to pressure, this diaphragm is indicating pressure. These switches are used to sense pressure of air, water or oil, in industrial application. This switch operates, when pressure of system is increase or decrease from set point.
3. ***Temperature switch:*** This type of switches consists of temperature sensing devices like RTD (resistance temperature device). This switch operates according to the value of measured temperature.

4. ***Toggle switch:*** This type of switch is commonly used in household application to ON and OFF electrical appliances. It has a lever by which we can move up or down to ON and OFF appliances.
5. ***Rotary switch:*** This type of switch is used to connect one line with one of the many lines. Nob of multi-meter, channel selector, range selector metering device band selector in communication devices are the examples of this type of switch. This switch is same as single pole multi throw switch. But the arrangement of this switch is different.

Electrical Switches:

Electrical switches are nothing but it is a semiconductor device. These switches are more useful because of their low cost, small size and reliability. In this switch, used semiconductor materials like silicon (Si), germanium (Ge) etc. Generally, this **type of switches** is used in integrated circuits (ICs), electrical motor drives, HVAC application and also widely used as digital output (DI) of controller.

1. Relay
2. Bipolar transistor
3. Power diode
4. MOSFET
5. IGBT
6. SCR
7. TRIAC
8. DIAC
9. GTO

1) Relay: Relay works on the principle of electromechanical, so this switch is also known as electromechanical switch. When current pass through a coil, it will create a magnetic field around the coil. This amount of magnetic field depends on the amount of current passes through the coil. Arrangement of contacts is done such a way that, if current is increased with curtain limit contacts are energised and change its position. Sometimes, relay uses bi-metallic strip to sense the temperature for safety purpose. Relay are available in wide range of voltage and current. In power system, relay plays an important role in fault identification. In industries also, relays are used as a protecting device. Check the complete Working of Relay here.

2) Bipolar Transistor: bipolar junctions transistor has three terminals; base, emitter and collector. Transistors are work on three regions; cut-off, saturation and active region. Symbol of transistor is as shown in figure-6. For switching purpose, active region is not used. If sufficient amount of current is available at base terminal, transistor enter in to saturation region and current will flow through collector-emitter path and transistor act as a ON switch. If base current is not sufficient, circuit is open and current cannot flow through the collector-emitter and transistor enters in to cut-off region. In this region, transistor act as OFF switch. Transistor are used as an amplifier in electronics application and it also used to make a gate like AND, NOT in digital circuits and transistor is also used as a switching device in integrated circuit. Transistors are not useful in high power application because it has more resistive loss compared to MOSFET.

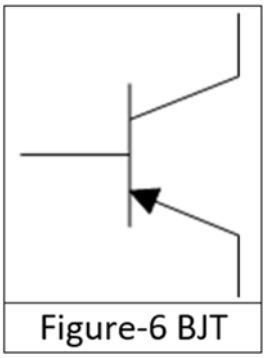


Figure-6 BJT

3) Power diode: Power diode have two terminals; anode and cathode. Diode is made up of p and n type semiconductor material and make pn-junction, which is known as diode. Symbol of power diode is as shown in figure-7. When diode is in forward bias current can flow through the circuit and in reverse bias blocks current. If anode is positive with respect to cathode, diode is in forward bias and act as a switch ON. Similarly, if cathode is positive with respect to anode, diode is in reverse bias and act as a switch OFF. Power diodes are used in power electronics application like, rectifier, voltage multiplier circuit and voltage clamper circuit, etc.

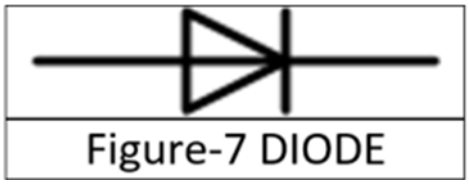


Figure-7 DIODE

4) MOSFET: MOSFET-Metal Oxide Semiconductor Field Effect Transistor. MOSFET has three terminals; gate, drain and source. MOSFET is works on two basic forms; Depletion type and Enhancement type. If gate-source voltage (V_{GS}) is not sufficient, MOSFET works as depletion type and depletion mode of MOSFET is similar to OFF switch. If gate-source voltage (V_{GS}) is sufficient, MOSFET works as enhancement type and enhancement mode of MOSFTE is similar to ON switch. Range of switching of MOSFET is tens of neon seconds to a few hundred microseconds. MOSFET used in linear voltage regulator, chopper and audio frequency power amplifier, etc. Check here for MOSFET Circuits.

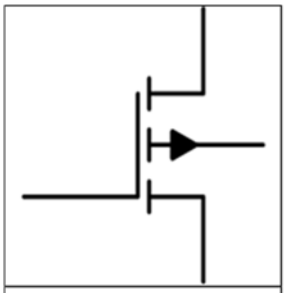


Figure-8 MOSFET

5) IGBT: IGBT- Insulated Gate Bipolar Transistor. IGBT is a combination of BJT and MOSFET. IGBT has a high input impedance and high switching speeds (characteristic of MOSFET) as well as low saturation voltage (characteristic of BJT).

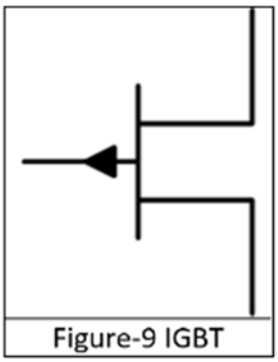


Figure-9 IGBT

IGBT has three terminals; Gate, Emitter and Collector. IGBT can control with the use of gate terminal. It can be switched ON and OFF by triggering and disabling its gate terminal. IGBT can block both positive and negative voltage same as GTO. IGBT is used in inverter, traction motor control, induction heating and switched mode power supplies.

6) SCR: SCR- Silicon Controlled Rectifier. SCR has three terminals; Gate, Anode and Cathode. Working of SCR is same as diode, but SCR start conduction when it is in forward bias (cathode is negative and anode is positive) and positive clock pulse at the gate is also required. In forward bias, if clock pulse of gate is zero, SCR turned off by forced commutation and in reverse bias SCR is remains in OFF state same as diode. SCRs are used in motor control, power regulators and lamp dimming.

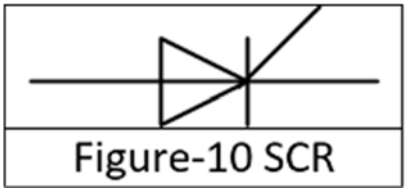


Figure-10 SCR

7) TRIAC: TRIAC is same as two SCRs connected in inversely parallel with gate connected. TRIAC is bi-direction device. TRIAC has three terminals; Main terminal 1 (MT), Main terminal 2 (MT2) and gate. MT1 and MT2 terminals are connected with circuit which we want to control and gate is available for triggering pulse by positive voltage or negative voltage. When MT2 terminal is at positive voltage with respect to MT1 terminal and gate is also positive triggered, then SCR-1 of TRIAC triggers. When MT1 terminal is at positive voltage with respect to MT2 terminal and gate is also positive triggered, then SCR-2 of TRIAC triggers. TRIAC can used for both sources AC and DC but generally, TRIAC is used in AC application like motor control, switching in lights (industrial and domestic), etc. Check here for Triac Dimmer Circuit.

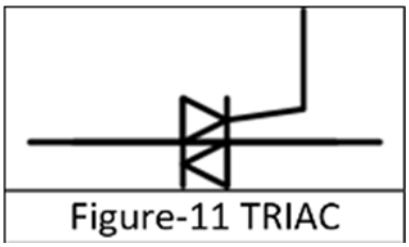


Figure-11 TRIAC

8) DIAC: DIAC- Diode AC switch. DIAC has two terminals. This switch can operate in both direction. Symbol of DIAC is as shown in figure-12. DIAC works on two regions; forward blocking or reverse blocking region and avalanche breakover region. When applied voltage is less than breakover voltage DIAC works in forward blocking or reverse blocking region. In this region DIAC act as OFF switch. When applied voltage is greater than breakover voltage, avalanche breakdown occurs and DIAC act as ON switch. DIAC cannot switch sharply for low voltage and low current application as compared to TRIAC and SCR. DIAC used in light dimming, control of universal motor and heat control circuit.

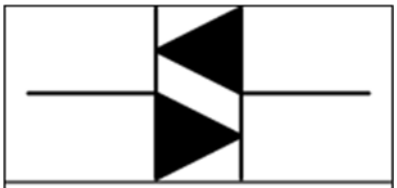


Figure-12 DIAC

9) Gate Turn-off Thyristor: GTO has three terminals; Gate, Anode and Cathode. As name suggest, this device can turn OFF through gate terminal. In symbol of GTO consists of two arrows on the gate terminal, which shows the bidirectional flow of current through the gate terminal.

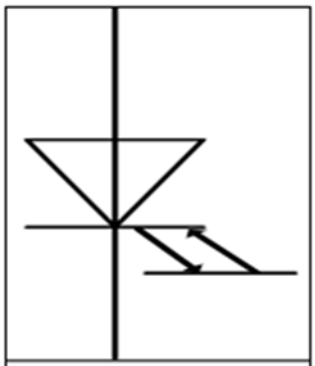


Figure-13 GTO

This device can turn ON by applying a small positive gate current and turn OFF by negative pulse from the gate terminal. GTO used in inverters, AC & DC Drives, induction heater and SVC (static VAR compensation). GTO cannot use for turning inductive loads off, without the help of the snubber circuit.

Program Outcomes (POs)

PO1	**Engineering Knowledge:** Apply the knowledge of mathematics, science, and engineering fundamentals to solve the complex electrical engineering problems.
PO2	**Problem Analysis:** Identify, formulate, review research literature, and analyze complex Electrical and Electronics Engineering problems enabling attainment of conclusions using first principles of mathematics, natural sciences, and engineering sciences.
PO3	**Design/Development of Solutions:** Design solutions, components or process for complex Electrical Engineering problems to meet the specified needs considering public health, safety and environmental considerations.
PO4	**Conduct Investigations of complex problems:** Exercise research knowledge and technical methodology for design, analysis and interpretation of data to converge to a suitable solution.
PO5	**Modern Tool Usage**: Use modern engineering tools, softwares and equipments to predict, analyze and model engineering problems.
PO6	**The Engineer & Society:** Apply reasoning skills to assess societal, health, safety, legal and cultural issues relevant to the professional engineering practice and take consequent responsibilities in the society
PO7	**Environment and Sustainability:** Realize the impact of the professional engineering solutions and demonstrate the knowledge for sustainable development in environmental context
PO8	**Ethics:** Apply and realize the professional ethics and responsibilities in Electrical engineering practice.
PO9	**Individual and Team Work:** Exhibit Individuality, Leadership and Team spirit in multidisciplinary settings.
PO10	**Communication:** Communicate, comprehend, write reports, design documentation and presentation effectively on complex engineering activities
PO11	**Project Management & Finance:** Demonstrate the Electrical engineering and management principles adhering to financial strategies to manage projects as a member or leader in a team
PO12	**Life Long Learning:** Inculcate independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes (PSOs)

PSO 1: Electrical drives and control: Graduates will Analyze, design and provide Engineering solutions in the field of Power Electronics and Drives

PSO 2: Embedded system: Graduates will Simulate, experiment and solve complex problems in Embedded System.

KSR INSTITUTE FOR ENGINEERING AND TECHNOLOGY

VISION

To become a globally recognized Institution in Engineering Education, Research and Entrepreneurship.

MISSION

- Accomplish quality education through improved teaching learning process.
- Enrich technical skills with state of the art laboratories and facilities.
- Enhance research and entrepreneurship activities to meet the industrial and societal needs.

Department of EEE

VISION

To produce world class Electrical and Electronics Technocrats and Entrepreneurs with social responsibilities.

MISSION

- Impart quality education in the field of Electrical and Electronics Engineering through state of the art learning ambience.
- Enrich interdisciplinary skills and promote research through continuous learning.
- Enhance professional ethics, entrepreneurship skills and social responsibilities to serve the nation.

BEES Magazine

February 2017