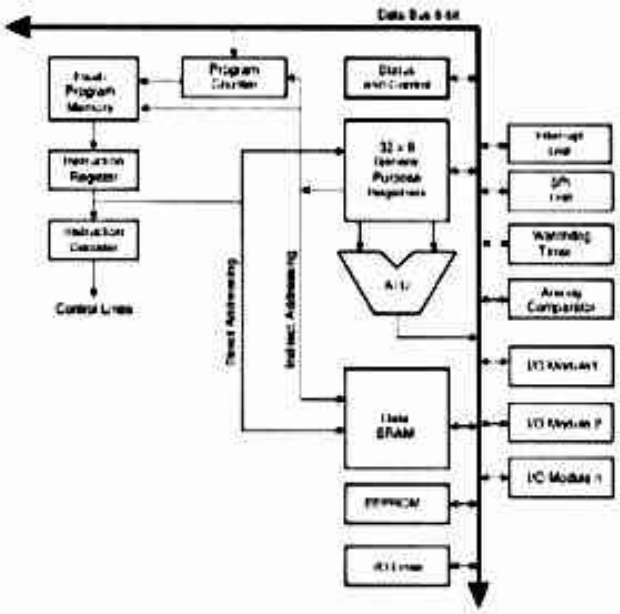
Why learn AVR Architecture?

Learning architecture of the micro-controller gives deep understanding of how it works. It also enables in writing better software for the same. Also programmers who understand the underlying hardware tend to write better programs.
The entire tutorial series will be based around **Atmega32** MCU from Atmel. Hence in this tutorial we will look at AVR architecture in general however all the specifics would point to Atmega32.

Block Diagram

Fig below shows the architecture of the MegaAVR series of controllers. As with any MCU ALU forms the core of the controller. A typical AVR Mega Series MCU has following hardware units inbuilt:

- The ALU
- 32 General Purpose Registers
- Special Function Registers
- Static RAM
- EEPROM
- Flash
- Timer/Counter
- Compartor
- Watch Dog timer
- Protocols: UART, SPI, I2C
- Interrupt Handler
- IO Ports
  - Analog
  - Digital



The ALU of an AVR MCU process data 8 bits at a time. It performs the
arithmetic and logical operations. The data for the ALU comes from the registers. If you wish to understand how an ALU works go through this ALU in Detail tutorial. Now instead of going into details of all other units described above, let me give you a analogy with 'C' program that you write and execute on a desktop computer.

Let's say we write a simple program like below:

```
1.  #include<stdio.h>
2.  int count = 0;
3.  void main()
4.  {
5.  for( i = 0; i<100; i++)
6.  {
7.  printf("hello world %d", i);
8.  }
9.  }
```

You might be saying where did this thing come from now? Exactly, lets put this into context. There are lot of things that the compiler and the accompanying software takes care of, when you write a similar programs on your computer. Things like **printf** will display the content and the string on monitor. You really need not care about how the system does it. Neither you'll have to worry about where the variables like i will be stored. Not that you'll have to know all of this for writing AVR programs. However understanding what goes where will help you a long way. Now that I have put things in air, let me show you a simple blinky program on AVR.

32 General Purpose Registers

Special Function Registers

Static RAM

EEPROM

Flash

Timer/Counter

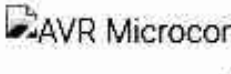
Compartor

Watch Dog timer

Protocols: UART, SPI, I2C

Interrupt Handler

IO Ports


AVR Microcontroller

AVR Microcontroller

There are a number of standard microcontroller families which are used in various applications as per their competence and possibility to perform the preferred task, most common microcontrollers are 8051, AVR and PIC microcontroller. In the year 1996, AVR was developed by Atmel Corporation. The architecture of this microcontroller was developed by "Alf-Egil Bogen" and "Vegard Wollan". The term AVR derives from its developers and stands for Alf-Egil Bogen Vegard Wollan RISC microcontroller, also known as "Advanced Virtual RISC". The first microcontroller AT90S8515 was based on the AVR architecture, but the first microcontroller to hit the salable market was AT90S1200 in the year 1997. In this article, we will present you to this basic of microcontroller, and its applications.

What is AVR Microcontroller?

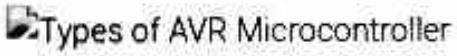
AVR microcontrollers are very popular, used in numerous applications, particularly in project prototyping and also in embedded devices. This microcontroller is an 8-bit RISC (Reduced Instruction Set Computing) architecture microcontroller in the market since 1996 which is having SRAM, on-chip programmable flash memory, IO data space, and the EEPROM. This is the first microcontroller in the marketplace, which has on-chip flash storage.

AVR microcontrollers are obtainable in three categories

Tiny AVR: This microcontroller has Less memory, small in size, apt only for simpler applications.

Mega AVR: This microcontroller is the most popular ones having a good amount of memory up to 256KB, higher no. of inbuilt peripherals and fit for modest to difficult applications.

Xmega AVR: This microcontroller is used commercially for compound applications, which need large program memory and also high speed.


Types of AVR Microcontroller

Types of AVR Microcontroller

AVR microcontroller performs most of the commands in the single execution cycle. These are around 4 times faster than PIC microcontrollers, they consume low power and can be worked in different power saving modes. The following table shows the comparison between the three most commonly used microcontrollers namely, 8051, PIC and AVR.

AVR is an 8-bit microcontroller appropriate to the family of RISC. In this architecture, the instruction set of the computer are not only less in number but also faster and simpler in operation. The other kind of classification is CISC.

Series Name	Pins	Flash Memory	Special Feature
Tiny AVR	6-32	0.5-8 KB	Small in size
Mega AVR	6-32	4-256KB	Extended peripherals
Xmega AVR	44-100	16-384KB	DMA, Event System included

AVR Microcontroller Architecture

The architecture of the AVR is shown below, it uses a "Harvard architecture" thus it has separate buses and memories for data and program. Instructions in the program memory are performed with a single level pipelining. While one instruction is being achieved, the subsequent instruction is pre-fetched from the program memory. This thought allows instructions to be performed in every CLK cycle and that suggest AVR runs at around 1 MIPS / MHz

AVR Architecture

CPU

The CPU of the AVR microcontroller is same but so simple like the one in a computer. The main purpose of the CPU is to confirm correct program performance. Therefore, the CPU must be able to access perform calculations, memories, control peripherals & handle interrupts. The CPUs of Atmel's 8-bit and 32-bit AVR are based on an innovative "Harvard architecture" thus every IC has two buses namely one instruction bus and data bus. The CPU reads executable instructions in instruction bus, wherein the data bus, is to read or write the corresponding data. The CPU core of the AVR consists of the ALU, General Purpose Registers, Program Counter, Instruction Register, Instruction Decoder, Status Register and Stack Pointer

Flash Program Memory

The program of the AVR microcontroller is stored in non-volatile programmable Flash program memory which is just similar to the flash storage in your SD Card or Mp3 Player. The Flash program memory is separated into two units. The first unit is the Application Flash section. It is where the program of the AVR is stored. The second section is named as the Boot Flash section and can be fixed to perform directly when the device is powered up. One significant fact to note is that the microcontrollers Flash program memory has a resolution of at least 10,000 writes/erase cycles.

SRAM

The SRAM (Static Random Access Memory) of the AVR microcontroller is just like computer RAM. While the registers are used to execute calculations, the SRAM is used to supply data through the runtime. This volatile memory is prearranged in 8-bit registers.

EEPROM

The term EEPROM stands for Electronically Erasable Read-Only Memory is like a nonvolatile memory, but you can't run a program from it, but it is used as long time storage. The EEPROM doesn't get removed when the IC loses power. It's a great place for storing data like device parameters and configuration of the system at runtime so that it can continue between resets of the application processor. One significant fact to note is that the EEPROM memory of the AVR has a limited lifetime of 100,000 writes / EEPROM page – reads are limitless. Keep this in mind in your application and try to keep writing to a minimum, so that you only write the small amount of info required for your application every time you update the EEPROM.

Digital I/O Modules

The digital I/O modules let digital communication or logic communication with the AVR microcontroller and the exterior world. Communication signals are that of TTL/CMOS logic.

Analog I/O Modules

Analog I/O modules are used to input or output analog information from or to the exterior world. These modules comprise analog comparators and analog-to-digital converters (ADC).

Interrupt Unit

Interrupts have enabled the microcontroller to monitor particular events in the background while performing and application program & respond to the occurrence if required pausing the unique program. This is all synchronized by the interrupt Unit.

Timer

Most AVR microcontrollers have at least one Timer or Counter module which is used to achieve timing or counting operations in the microcontroller. These comprise time stamping, counting events, measuring intervals, etc.

Watchdog

All AVR microcontrollers have an internal Watchdog Timer. It has very limited useful features comprising: distinct 128kHz CLK source, skill to reset the microcontroller and produce an interrupt.

USART / SPI / I2C

The units like USART or SPI or I2C are used for serial communication with the exterior world. An instance is the USART peripheral which uses the RS232 standard.

Thus, this is all about a tutorial on AVR microcontroller, its basics, and architecture.We hope that you have got a better understanding of this concept. Furthermore, any doubts regarding of this concept or to implement any microcontroller based projects. Please give your valuable suggestions by commenting in the comment section below. Here is a question for you, What are the applications of AVR microcontroller?

Table of Contents

PIC MICROCONTROLLER ARCHITECTURE: PIC stands for Peripheral Interface Controller. PIC microcontroller was developed by microchip technology in 1993. It was developed for supporting PDP computers to control its peripheral devices and that's why it was named Peripheral Interface Controller. PIC microcontrollers are of low cost, very fast and easy for the programming and execution of program. Their interfacing with other peripherals is also very easy. PIC Microcontrollers from Microchip Company are divided into 4 large families. In this PIC MICROCONTROLLER ARCHITECTURE article, I will explain step by step about PIC MICROCONTROLLER ARCHITECTURE and components used in pic microocntrollers. I recommend you to check a list of **Pic microcontroller project** here.

- First family: PIC10 (10FXXX) called Low End
- Second family: PIC12 (PIC12FXXX) called Mid-Range
- Third family: PIC16 (16FXXX)
- Fourth family: PIC 17/18 (18FXXX)

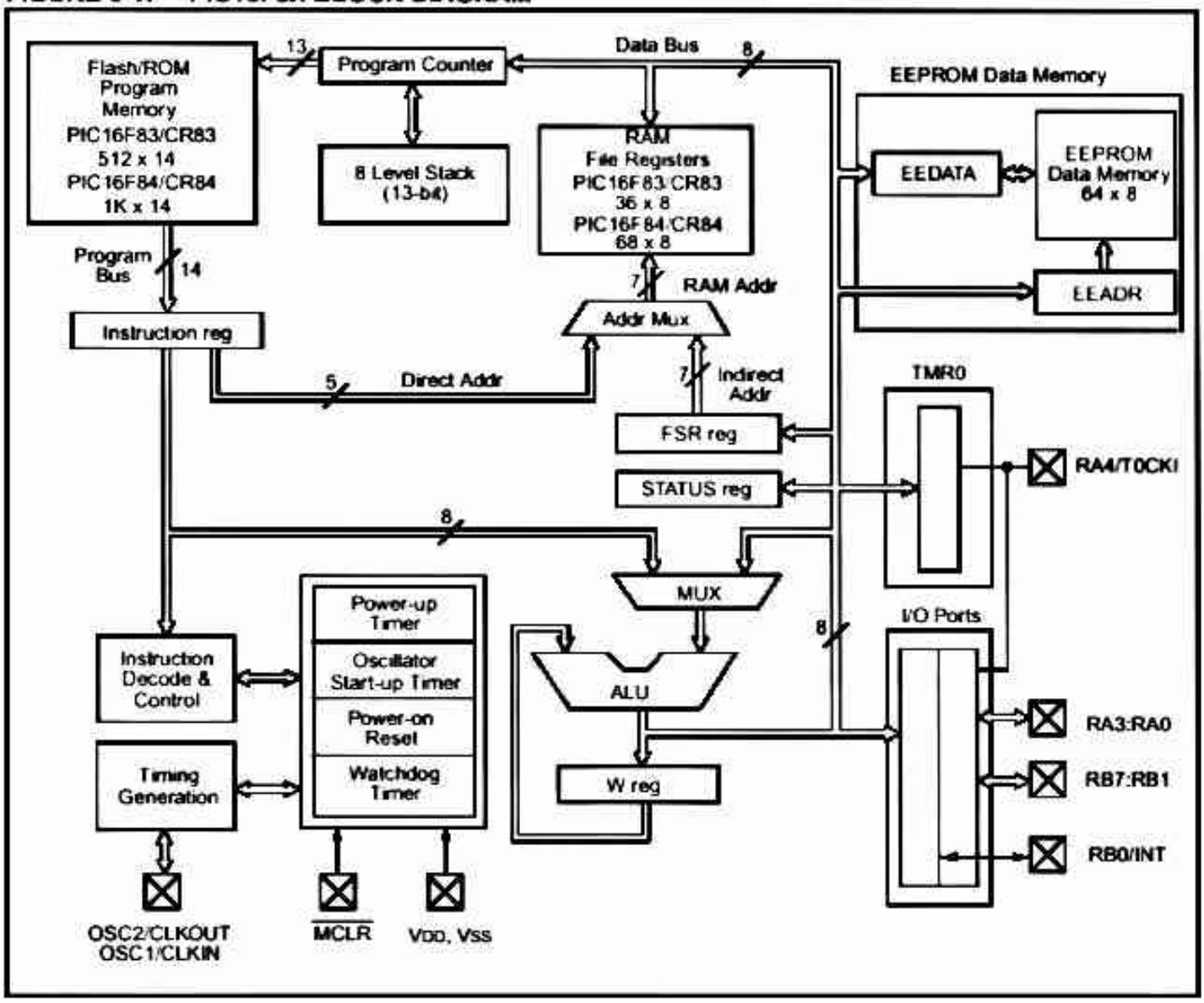
Each family has a variety of components along with built in special features. It offers a lot of memory sizes and pin packages and different clock ratings.

ARCHITECTURE:

PIC Microcontroller architecture is based on Harvard architecture and supports RISC architecture (Reduced Instruction Set Computer). PIC microcontroller architecture consists of memory organization (ram, rom, stack), CPU, timers, counter, ADC, DAC, serial communication, CCP module and I/O ports. PIC microcontroller also supports the protocols like CAN, SPI, UART for interfacing with other peripherals.

PIC MICROCONTROLLER ARCHITECTURE block diagram

FIGURE 3-1: PIC16F8X BLOCK DIAGRAM



1. **CPU (Central Processing Unit):**

PIC microcontroller's CPU consists of

- Arithmetic logic unit (ALU)
- Memory unit (MU)
- Control unit (CU)
- Accumulator

ALU is used for arithmetic operations and for logical decisions. Memory is used for storing the instructions after processing. Control unit is used to control the internal and external peripherals which are connected to the CPU and accumulator is used for storing the results.

2. **MEMORY ORGANIZATION:**

PIC microcontroller memory module consists of mainly 3 types of memories:

- **PROGRAM MEMORY:**

It contains the written program after we burned it in microcontroller. Program Counter executes commands stored in the program memory, one after the other. Pic microcontroller can have 8K words x 14 bits of Flash program memory that can be electrically erased and reprogrammed. Whenever we burn program into the micro, we erase an old program and write a new one.

- **DATA MEMORY:**

It is a RAM type which is used to store the data temporarily in its registers. The RAM memory is classified into banks. Each bank extends up to 7Fh (128 bytes). Number of banks may vary depending on the microcontroller. PIC16F84 has only two banks. Banks contain Special Function Registers (SFR) and General Purpose Registers (GPR). The lower locations of each bank are reserved for the Special Function Registers and upper locations are for General Purpose Registers.

General Purpose Registers (GPR):

These registers don't have any special function. These are used for general purpose for multiplying, addition or subtraction and then storing the results in other registers. CPU can easily access the data in these registers.

Special Function Registers (SFR):

These registers are used for special purposes and they cannot be used as normal registers. Their function is set at the time of manufacturing. They perform the function assigned to them and user cannot change the function of SFR. Three important SFRs for programming are:

STATUS register: It changes the bank

PORT registers: It assigns logic values 0 or 1 to the ports

TRIS registers: It is a data direction register for input and output

- **DATA EEPROM:**

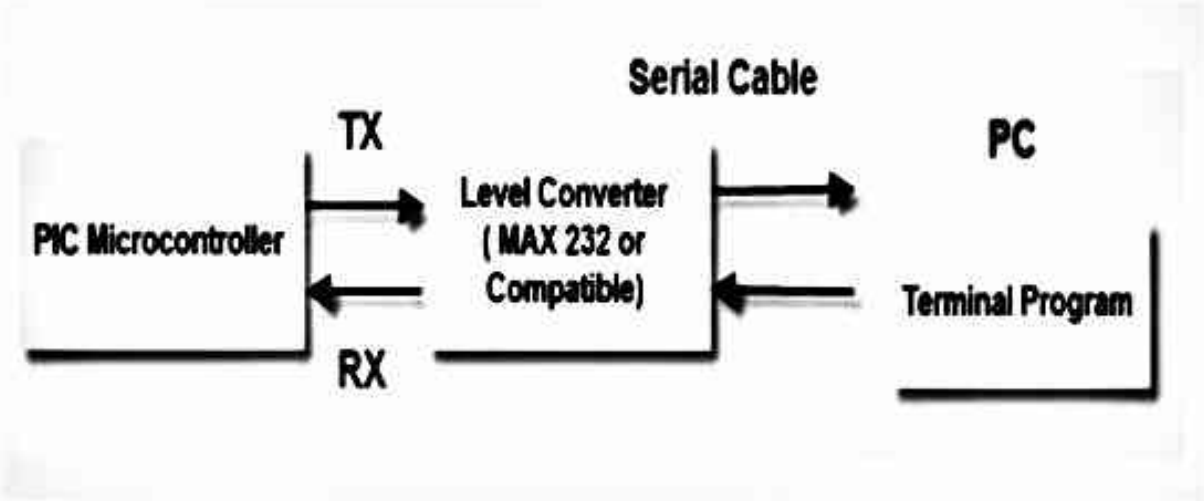
This memory allows storing the variables as a result of burning the written program. It is readable and writable during normal operation (over the full VDD range). This memory is not directly mapped in the register file. It is indirectly addressed through the SFRs. There are six SFRs which are used to read and write to this memory (EECON1, EECON2, EEDATA, EEDATH, EEADR, EEADRH).

3. **SERIAL COMMUNICATION:**

The transfer of one bit of data at time consecutively over a communication channel is called Serial Communication. There are three protocols of serial communication:

- **USART:** It stands for Universal synchronous and Asynchronous Receiver and Transmitter which provides a serial communication in two devices. In this protocol data is transmitted and received bit by bit through a single wire according to the clock pulses. To send and receive data serially the PIC microcontroller has two pins TXD and RXD.

- **SPI Protocol:** SPI stands for Serial Peripheral Interface. It is used to send data between PIC microcontrollers and other peripherals like sensors, shift registers and SD cards. Three wire SPI communications is supported in PIC microcontroller between two devices on a common clock source. SPI protocol has greater data handling capability than that of the USART.**I2C Protocol:** I2C stands for Inter Integrated Circuit, and this protocol is used to connect low speed devices like microcontrollers, EEPROMS and A/D converters. PIC microcontroller support two wire Interface or I2C communication between two devices which can work as both Master and Slave device.



Serial Communication

4. **INTERRUPTS:**

There are 20 internal interrupts and three external interrupt sources in PIC microcontrollers which are related with different peripherals like ADC, USART, Timers, and CCP etc.

5. **I/O PORTS:**

Let us take PIC16 series, it consists of five ports, such as Port A, Port B, Port C, Port D and Port E.

- **Port A:**This port is 7-bit wide and can be used for both input and output. The status of TRISA register decided whether it is used as input or output port.
- **Port B:**It is an 8-bit port. This port also can be used as input and output. Moreover in input mode four of its bits are variable according to the interrupt signals.
- **Port C:**It is also an 8-bit port and can be used as both input and output port which is determined by the status of the TRISC register.
- **Port D:**This 8-bit port, unlike Port A, B and C is not an input/output port, but is used as acts as a slave port for the connection to the microprocessor When in I/O mode Port D all pins should have Schmitt Trigger buffers.
- **Port E:**It is a 3-bit port which is used as the additional feature of the control signals to the A/D converter.

6. CCP MODULE:

A CCP module works in the following three modes:

- **Capture Mode:** In this mode time is captured when a signal is arrived, or we can say that, when the CCP pin goes high it captures the value of the Timer1.
- **Compare Mode:** It works same as an analog comparator, which means that when timer 1's value reaches some reference value it will give an output signal.

6. CCP MODULE:

A CCP module works in the following three modes:

- **Capture Mode:** In this mode time is captured when a signal is arrived, or we can say that, when the CCP pin goes high it captures the value of the Timer1.
- **Compare Mode:** It works same as an analog comparator, which means that when timer 1's value reaches some reference value it will give an output signal.
- **PWM Mode:** This mode provides a 10 bit resolution pulse and duty cycle that is programmable.

7. Timers:

Timers and counters are important as timers can tell the time and count. PIC microcontroller can have up to four timers (depending upon the family) Timer0, Timer1, Timer2 and Timer3. Timer0 and Timer2 are of 8-bits while the Timer1 and Timer3 are of 16-bits, which can also be used as a counter. These timers work according to the selected modes.

8. D/A CONVERTER:

There are no **analog outputs in PIC Microcontroller.** To get analog output we have to use external Digital-to-Analog Converter (DAC). It can convert 8 bits of digital number from the eight digital outputs of PIC microcontroller.

Robots – Types & Applications

History of Robotics

History:

This segment is deliberated to offer you with a summary of the history of robotics. As you may have presumed, the robotics history is entangled with the history of science, technology and the fundamental principles of progress. Technology employed in electronics, computers, even pneumatics & hydraulics can all be measured as a fraction of the robotics history. Robotics at present symbolizes one of the mankind's supreme achievements and is the only best endeavor of mankind to create an artificial, electronic being.

Though robots are regarded as a 20th century discovery, their origins lie in the far history. From the initial time, public have shaped myths regarding automatic beings built-in their individual likeness with extraordinary human powers. The prehistoric age around 270BC Greeks & Egyptians manufactured mechanical machines to execute easy tasks. In modern times, automatic toys amuse and ever more complex machinery was invented. The thought of a realistic motorized humanoid monster named as "Frankenstein" in the year 1818 surveys what occurs when a man-made giant is gifted life by a knowledgeable scientist (Dr. Frankenstein). As the advancement in the computer technology progressed at a great pace, scientists became more fascinated in construction of intellectual machines that can ultimately have some logic to work themselves. At present, robots of all types occupy our globe and are brought into play for diverse applications in

- In the year 1921, the Czech dramatist "Karel Capek" coins the world by using the word robot in his play Rossum's Universal Robots (R.U.R). This word robot is derived from a Czech word which means "compulsory labor."
- "Runaround" was composed by Asimov about robots in the year 1942, it held the "Three rules for robots"
  - Robots are not harmful to the humans, or through working, permit a human to come and damage.
  - A robot must follow the commands given by human beings apart from where such instructions would conflict with the First Law of Robotics.
  - A robot must defend its own survival providing such safety does not clash with the First and the Second Law of Robotics.
- In the year 1956, George Devol and Joseph Engelberger established the first robot company.
- In the year 1959, computer assisted manufacturing was verified at MIT.
- UNIMATE- The first industrialized robot was online in a General Motors automobile plant, in the year 1961.
- 1963 was a revolutionary year, first computer controlled robotic arm was designed and it was named as Rancho Arm. The invention was basically for the handicapped peoples.

The inventions in the field of Robotics were never ending and gave human beings a sudden surprising gift as & when launched. After Rancho's Arm various other inventions too were done, but all of the above was the first among all.

Different Branches Occupied in the Develop ment of Robotics:

Robotics in contrast to other branches is a reasonably new domain of engineering. It is a multi-disciplinary domain. The different branches occupied in the development of Robotics are:-

1. **Mechanical Engineering**: Deals with the machinery & structure of the Robots.
2. **Electrical Engineering**: Deals with the controlling & intelligence (sensing) of Robots.
3. **Computer Engineering**: Deals with the movement development and observation of Robots.

Classification of Robots:

Robots are categorized depending upon the circuits of the Robots and the variety of application it can perform. The robots are classified into three types:

- **Simple level Robots**- These are automatic machines which do not contain complex circuit. They are developed just to extend human potential. For Example- Washing Machine.
- **Middle level Robots**– These robots are programmed but can never be reprogrammed. These robots contain sensor based circuit & can perform multiple tasks. For Example- Fully Automatic Washing Machine.
- **Complex level Robots**- These robots are programmed and can be reprogrammed as well. They contain complex model based circuit. For Example- Laptop or Computer.

Types of Robotics:

Robotics is an area of interest to human beings for more than one hundred years. On the other hand, our perception over robots is influenced by the media and international film industry (Hollywood). You may ask- what robotics is all about? In my views, a robot's distinctiveness transforms depending on the atmosphere it works in. Some of these are as follows:-

1. **Outer Space** – Robotic arms that are under the control of a human being are employed to unload the docking cove of outer-space shuttles to launch satellites or to build a space station.
2. **The Intelligent Home** – Robotic systems can nowadays scrutinize home safety, ecological circumstances and energy consumption. Door & windows can be unlocked mechanically and electrical device such as lights and A/C can be pre-programmed to turn on. This helps residents to enjoy appliances irrespective of their mobility.
3. **Exploration** – Robots can enter the environments that are injurious to human beings. An illustration is observing the atmosphere within a volcano or investigating our deep marine life. NASA has utilized robotic probe for environmental study, ever since the early 60's.
4. **Military Robots** – Flying robot drones are brought into play for close watch in present time's modern armed force. In the future robotic airplane and automobiles could be employed to transmit petroleum, bullets, bombs, etc or clear minefields.
5. **Farms** – Programmed robots are used by harvesters to cut and collect crops. Robotic milk farms are existing permitting workers to nourish and milk their cattle distantly.
6. **The Car Industry** – Robotic arms are used, these arms are able to execute numerous tasks in the car manufacturing & assembling procedure. They carry out jobs such as sorting, cutting, welding, lifting, painting and bending. Similar functions but on a minor scale are now being intended for the food industry to execute tasks like- the trimming, cutting and processing of different types of meats like- chicken, beef, fish, lamb, etc.
7. **Hospitals** – The development of a robotic suit is under construction that will allow nurses to raise patients without injuring their backbones. Scientists in Japan have crafted a power facilitated suit which will provide nurses the additional power that they need to lift patients.
8. **Disaster Areas** – Observation robots built-in with superior sensing and imaging gears. This robot can work in dangerous environments like urban site spoiled by earthquakes by inspecting floors, walls, and roofs for structural reality.
9. **Entertainment** – Interactive robots that shows behaviors and education capability. One such robot is owned by SONY which moves around freely, responds to all your commands, carries your luggage and even responds to your oral instructions.

This is not the end of Robotic world; there is many more application of Robotics.

Applications:

Currently, robots perform a number of different jobs in numerous fields and the amount of tasks delegated to robots is rising progressively. The best way to split robots into types is a partition by their application.

1. Industrial robots – These robots bring into play in an industrialized manufacturing atmosphere. Typically these are articulated arms particularly created for applications like- material handling, painting, welding and others. If we evaluate merely by application then this sort of robots can also consist of some automatically guided automobiles and other robots.

2. Domestic or household robots – Robots which are used at home. This sort of robots consists of numerous different gears for example- robotic pool cleaners, robotic sweepers, robotic vacuum cleaners, robotic sewer cleaners and other robots that can perform different household tasks. Also, a number of scrutiny and tele-presence robots can also be considered as domestic robots if brought into play in that sort of environment.

3. Medical robots – Robots employed in medicine and medicinal institutes. First & foremost surgical treatment robots. Also, a number of robotic directed automobiles and perhaps lifting supporters.

4. Service robots – Robots that cannot be classed into any other types by practice. These could be various data collecting robots, robots prepared to exhibit technologies, robots employed for research, etc.

5. Military robots – Robots brought into play in military & armed forces. This sort of robots consist of bomb discarding robots, various shipping robots, exploration drones. Often robots at the start produced for military and armed forces purposes can be employed in law enforcement, exploration and salvage and other associated fields.

6. Entertainment robots – These types of robots are employed for entertainment. This is an extremely wide-ranging category. It begins with model robots such as robosapien or the running photo frames and concludes with real heavy weights like articulated robot arms employed as movement simulators.

7. Space robots – I would like to distinct out robots employed in space as a split apart type. This type of robots would consist of the robots employed on Canadarm that was brought into play in space Shuttles, the International Space Station, together with Mars explorers and other robots employed in space exploration & other activities.

Embedded Design Concepts

As embedded processors become more powerful and cost effective, it is becoming increasingly common for an embedded processor to replace analog or mechanical timing and control circuits. A similar influence may be also seen in new designs. Embedded processors are replacing functions in new designs that have been traditionally implemented by hardwired circuits, allowing design features to be software upgradeable.

Our goal at Embedded Design Concepts is to promote the use of embedded processors in new or existing electronic circuit designs. We are dedicated to offering the design support, design tools and embedded components needed to bring your embedded design from initial concept to production release.

Consulting Services

We offer consulting services that range from full system design to help in critical areas of embedded design.

- Custom embedded design service

  Do you have an idea for a product that could utilize an embedded microcontroller, but you are not sure where to start? Contact us and we will help with the design process.

- Existing design modification

  Would your existing design benefit from an embedded device? Most embedded designs result in a lower factory cost due to reduced component count. Embedded designs also allow the used of firmware upgrades to include additional features without changes to the existing hardware. Contact us if you would like to evaluate or incorporate an embedded micro-controller into your existing design.

- Software

  We specialize in developing critical software drivers used for A/D conversion, serial and keypad input, LCD display, closed loop control, and PWM or phase control output

- PCB layout service

  Do you have a working prototype that is ready for PCB layout? Contact us and we will work with you to produce a high quality PCB for your design.

- Device programming service

  Need to program a device that your current programmer doesn't support? We offer programming services for many OTP, EPROM, EEPROM and FLASH based memory and micro-controller devices. We also offer low cost programmers for commonly used devices.

Design Tools

Whether a student, a hobbyist or a professional, some hesitate to incorporate an embedded device into a design when the cost of a universal programmer is considered. For most individuals and small businesses it is difficult to justify the additional expense of a universal programmer even if several products are designed around a common platform.

Embedded Design Concepts offers device programmers and development systems that have been designed to be affordable tools for embedded systems development. As an alternative to a single universal programmer model, we offer multiple programmers that target individual device families that are most commonly used in embedded designs. Our programmers offer a low cost alternative to a universal programmer for those who wish to pursue embedded system development on a budget.

- Device Programmers
    - 24Cxx Serial EEPROM programmer

      This programmer may be used to program 24xx series I^2C serial EEPROMS.
- Development Systems
    - 16F628 PIC Development Board

      This board may be used to prototype systems based on the PIC 16F628 microcontroller. Similar 18 pin PIC devices are supported as well. It includes provisions for crystal, ceramic resonator, RC or internal oscillator circuits, RS-232 communications support, and a 12 by 20 grid for additional circuitry.

Embedded Microcontroller Components

Looking for a particular microcontroller? We have the following microcontrollers available in stock. These are remaining stock from past or current projects. Most are unused with a few being samples that never found thier way into prototypes.

- Advanced Micro Devices (AMD) Microcontrollers

  D87C51
- Atmel Microcontrollers

  AT87F51 and AT89C52
- Intel Microcontrollers

  8031, 8035, 80C35, 80C39, 8048, 80C49, 80C196, 87C51FA, and 87C550
- Motorolla Microcontrollers

  68HC11A1P, 68HC11A1FN, XC68HC711E9, 68HC11E1FN, and X68HC912B32
- Microchip PIC microcontrollers

  12F675, 16F628, 16F684, 18F252 and 18F452
- Signetics Microcontrollers

  SCC2692AE1, SCN8031HCCN40, 87C751CCF40, 87C654-ZB2, and 87C654-ZB4
- Siemens Microcontrollers

  SAB-C501G-1RP (12Mhz 80C32 compatible)